Simulation of the Development of the Root System and associated Microbial Community of *Pinus radiata*

A thesis submitted in
partial fulfilment of the
requirements for the degree of
Doctor of Philosophy
in
Biological Systems Simulation

by
T.N. Brown

Lincoln University
1995

Abstract of a thesis submitted in partial fulfilment of the requirements for the Degree of Ph.D. in Biological Systems Simulation.

Simulation of the development of the Root System and associated Microbial Community of *Pinus radiata*

by T.N. Brown

Abstract

A simulation scheme to model three-dimensional plant root architecture and the location, growth and interaction of associated microbial communities has been developed. The scheme expands on previous root architecture models by using the mature root system morphology observed in the field as a spatial envelope to model the system's temporal development. The three-dimensional representation allows a uniquely detailed treatment of the spatial development of microbial populations and their interactions with the root system and with each other. Root morphology and microbial populations are described by different types of "node", each node records a position in three-dimensional space and other details specific to the feature it represents. For example, "Branch" nodes indicate the start of a higher order root, and "Fungal" nodes increase or decrease the level of a fungal population on a root's surface. A large number of probability density functions are required to produce the list of nodes that describe each root. A four-dimensional matrix is used as a convenient abstraction for these functions, the dimensions represent tree type, root order, a "feature" such as branching or microbial infection, and an "attribute" such as length or lifespan. Algorithms for the generation and manipulation of the node lists are given. Specific implementation issues, such as rapid location of roots within the area potentially affected by disease lesion, and visualisation of the simulated root system, are also addressed. Validation of models of complex, irregular entities such as plant root systems is difficult, some techniques, including an objective root distribution index, have been developed and applied in this research. The simulation software includes extensions to allow input and editing of observed root system architectures to simplify the extraction of relevant parameters from such observations.

The *Pinus radiata* / armillaria root-rot pathosystem has been used extensively as a test case for the scheme. The spread of the disease armillaria between roots of seedlings and the control of that spread by antagonistic strains of the soil fungus genus *Trichoderma* have been successfully modelled. Simulated root architectures are consistent with those observed in the field, and patterns of fungal spread through simulated stands are very similar to those reported in the literature. While this research was initially prompted by the *Pinus radiata* / armillaria pathosystem, the problems solved, namely representation and manipulation of three-dimensional root architecture and root / microbial spatial interaction, are essentially the same for most plant root ecosystems. The scheme should be a useful tool for the development of biological control programmes, both in formulating an understanding of the dynamics of the ecosystem to be modified, and in optimising the timing and position of biological control agent application. Further development of the scheme should include a generalised root ecosystem description syntax, and a re-implementation of the simulation's internal structure in an object oriented language. This should result in a versatile framework for detailed simulation of plant root microbial ecosystems.

Contents

Abstract ii

Contents iii

List of Tables ix

List of Figures x

1 Introduction **1**

1.1 Increasing stand productivity 3

1.2 The case study system 4

1.2.1 *Pinus radiata* in New Zealand 4

1.2.2 The root system of *P. radiata* 4

1.3 The organisms and their relationships 7

1.3.1 *Armillaria spp.* as a pathogen of *P. radiata* 8

1.3.2 *Trichoderma spp.* as a biological control agent for *Armillaria spp.* . . 10

1.3.3 *Rhizopogon spp.* as a symbiont of *P. radiata* 11

1.3.4 Effect of *Trichoderma spp.* on *P. radiata.* 12

1.3.5 Interaction of *Trichoderma spp.* with *Rhizopogon spp.* 13

1.3.6 Interaction of *Rhizopogon spp.* with *Armillaria spp.* 13

1.4 Current status of the armillaria pathosystem 13

1.5 The root system structure 14

1.6 Objectives 19

2 Model requirements and Existing Models **20**

2.1 Flexibility 20

2.2 Investigation of diverse environmental factors 21

2.2.1 Root Architecture 21

2.2.2 Root physiological status 21

2.2.3 Fungal populations 22

2.2.4 Armillaria induced mortality 23

2.2.5 Root and fungal interaction 23

2.2.6 Thinning 24

2.2.7 Ground topography 24

2.3 Model requirements – Summary 25

2.4 Frameworks for spatial models of root ecosystems 25

2.4.1 Root growth physiology models 26

2.4.2 Root morphology characterisation 26

2.4.3 Root Oriented Geometric Representations (ROGRs) 28

2.4.4 Root Density Models (RDMs) 28

2.5 Existing root growth simulations 30

2.5.1 Douglas Fir root architecture model for a single fungal species 30

2.5.2 Simulation of cereal root system architecture 31

2.5.3 Root density approach 31

2.5.4 Length/branching ratios for canopy model 32

2.5.5 Discrete, stochastic vs. continuous models 32

3 Model Development **33**

3.1 Spatial envelope for temporal development 33

3.2 Root architecture – static components 34

3.2.1 Nodes 34

3.2.2 Records – information packets 36

3.2.3 Root diameter 38
3.3 Root architecture - dynamic model processing 38
3.3.1 Status record 38
3.3.2 Status filter 40
3.3.3 Analysis procedure 41
3.3.4 Parser 43
3.4 Simulated root system construction 47
3.4.1 Parameter matrix 47
3.4.2 Constructor 49
3.5 Representation of root physiology 51
3.6 Fungal behaviour 51
3.6.1 Nodes 51
3.6.2 Elongation 52
3.6.3 Free soil fungi and roots 53
3.7 Rhizomorphs 54
3.8 Fungal interaction 54
3.9 Infection 55
3.9.1 Infection score 55

4 Parameter evaluation **58**

4.1 Derivation of distributions and parameters for system geometry 58
4.2 Data acquisition 60
4.2.1 *Armillaria spp.* distribution in dead root systems 60
4.2.2 Distribution of sites susceptible to armillaria in *Pinus radiata* seedlings 61
4.3 Two year old *Pinus radiata* root systems: field observations 61
4.3.1 Layered excavation 61
4.3.2 Stereo Image Pairs 63
4.3.3 Radial distribution 65

4.3.4 Inter-tree root contact 65

4.4 Sources of data – Summary 66

5 Implementation **70**

5.1 Programming environment 70

5.2 Data representation 70

5.3 Parameter functions 73

5.4 Construction of the root system 73

5.4.1 Vector notation 73

5.5 Voxel Lattice 75

5.6 Analysis of the generated root system 79

5.6.1 Profile wall analysis 79

5.6.2 Core sample analysis 81

5.7 Representation of tree statistics 84

5.7.1 Human readable form 84

5.7.2 Computer interpretation 84

5.8 Ground topography 86

5.8.1 Digitising 89

5.9 Line segment within a sphere 90

5.10 Shortest line connecting two line segments. 91

5.11 Azimuth from coordinates 95

5.12 Trapezoidal distributions 96

6 Validation **98**

6.1 Root architecture 98

6.1.1 Radial distribution of lateral roots 98

6.1.2 Profile walls 102

6.1.3 Core sampling 104

7 Results **106**

7.1 Disease centre expansion 106

7.1.1 Observed behaviour 106

7.1.2 Model response 108

7.2 Thinning 113

7.2.1 Observed behaviour 113

7.2.2 Model response 113

7.3 Biological control application 122

7.3.1 Observed behaviour 122

7.3.2 Model response 122

8 Discussion **127**

8.1 Model response to trial situations 127

8.1.1 Disease centre expansion 127

8.1.2 Thinning 128

8.1.3 Biological control application 128

8.2 Application to the *Pinus radiata* / armillaria pathosystem. 128

8.3 Parameterisation and Validation 129

8.4 Visualisation 130

8.5 General notes 131

8.5.1 Lists vs. Recursion 131

8.6 Summary 132

9 Future Directions **133**

9.1 A generic tree description 133

9.2 Improved physiological submodels 134

9.3 Software development 135

9.3.1 Programming language 135

9.3.2 User interface 136

9.4 Possible applications 137
9.4.1 Other disease systems 137
9.4.2 Root competition 137
9.4.3 Root sampling strategies 138
9.4.4 Visualisation 138

Publications **139**

Acknowledgments **139**

A File formats **147**
A.1 Configuration file `config.rs` 147
A.2 Filter files 148
A.3 Ground height file `ground.rs` 149
A.4 Display colour file `rgb.rs` 150

B Program notes and instructions **151**
B.1 Compiling the program 151
B.2 Running the program 152
B.2.1 Entering commands 153
B.2.2 Simulating cross infection – an example 154
B.3 Viewing the demonstration images 157

C Parameter files for *Pinus radiata*. **158**
C.1 `rs_numb.c` 158
C.2 `config.rs` 170

D Root maps used for parameterisation **172**
D.1 Eight year old trees 173
D.2 Sixteen year old trees 183

List of Tables

2.1 Comparison of geometric and density approaches. 29

3.1 Root system variables. 48

3.2 Root variable matrix form. 49

3.3 Root physiological phases. 51

B.1 Keyboard controls in cursor mode 155

B.2 Keyboard controls in non-cursor mode 156

List of Figures

1.1 The *Pinus radiata* fungal ecosystem. 8

1.2 Mycorrhizal colonisation of *Pinus spp.* roots 12

1.3 Root system structure. 15

1.4 Two year old *Pinus radiata* root system *in situ*. 16

1.5 2 year old *Pinus radiata* root system. 17

2.1 Spatial distribution of inocula reduction. 24

2.2 Frameworks for spatial models of root ecosystems. 25

2.3 Root density data sets. 31

3.1 Translation from physical root system to list representation. 35

3.2 Angle / azimuth notation 36

3.3 Fungal status information. 40

3.4 Root system analysis. 43

3.5 Root system parser, example data. 44

3.6 Nodes required to delimit fungal lesion. 50

3.7 An attenuating fungal lesion. 52

3.8 Fungal lesion expansion. 53

3.9 Free soil fungi – spherical regions 57

4.1 16×6 circular grid. 59

4.2 Framework for excavation. 62

4.3 Roots in a 100 mm layer. 63

4.4 Three dimensional image capturing. 64
4.5 Three dimensional image. 64
4.6 Contact between first order laterals. 67
4.7 Rip line root contact. 68
4.7 Rip line root contact. 69

5.1 Root intersection detection by voxel lattice. 76
5.2 A voxel lattice. 77
5.3 Internode intersection with trench wall. 80
5.4 A soil core sample. 81
5.5 Intersection of line segment with cylinder. 83
5.6 A point's position on the ground topography grid. 87
5.7 Ground topography. 88
5.8 Entry of root system from digitised images. 90
5.9 Shortest route between two line segments. 92
5.10 Shortest route between two line segments 92
5.11 A trapezoidal distribution. 97

6.1 Overlapping and alignment of segments in "evenness" index. 100
6.2 Examples of (un)evenness index scores. 101
6.3 Trench intercept, plan and elevation. 103
6.4 Core sample. 105

7.1 Armillaria distribution in a lodgepole pine stand. 107
7.2 Spread of armillaria in a simulated stand 109
7.3 Thinning adjacent to a diseased root system. 113
7.4 Thinning prevents cross infection. 115
7.5 Thinning causes cross infection. 118
7.6 Simulated *Trichoderma spp.* biocontrol. 123

8.1 A computer generated root architecture, plan and elevation. 130

8.2 Internode processing order. 131

B.1 Shape of three-dimensional cursor 154

B.2 Demo simulation layout 156

Chapter 1

Introduction

Plant leaf diseases are usually spread by airborne spores, splash droplets, or insects. As such vectors can move rapidly over small to large distances there is little physical restriction on the frequency of contact of infectious material with plant surfaces. The critical factors controlling disease impact in above ground plant disease systems are host resistance, disease virulence, rate of growth and reproduction of the pathogen, and environmental conditions. The dominance of such factors is reflected in the models used to characterise such systems, usually sets of mathematical equations based on population and epidemiology functions.

With plant root diseases, however, the frequency of contact between pathogen and host is greatly reduced by the intervening soil matrix. Root diseases are usually spread by growth, some have vector organisms whose movement is also restricted, and a few have motile spores that can swim in moist conditions. The decrease in contact frequency is partly countered by an increase in host range; while many foliar pathogens are restricted to a single host species, many soil borne plant pathogens are able to attack most members of a genus, or even members of quite separate plant groups. But for any specific pathosystem, host / disease contact is still the controlling factor. The spatial aspects of the system, which control root contact with infectious material, are essentially defined by the morphology of the plant root system.

Although the development of this simulation was initially inspired by questions raised by the *Pinus radiata* / armillaria root rot pathosystem, the basic simulation problem (three-dimensional representation of root and fungal ecosystems) is largely independent of the specific behaviour of the species involved. So while the *Pinus radiata* / armillaria system is used extensively as a test

case, the real goal of this work is the specification of a three-dimensional plant-root microbial ecosystem simulation scheme, and an initial implementation of the scheme to test its ability to realistically simulate plant root ecosystem behaviour.

Pinus radiata has been the traditional choice for softwood production in New Zealand, and its complete dominance of New Zealand's forestry industry is reflected in the global distribution of *Pinus radiata* stocks : Chile 35%, New Zealand 34%, Australia 20%, Spain 7%, Other 4% [63]. In 1991, the forestry industry earned New Zealand around $1600M NZ, or 23% of income from primary products, New Zealand's main revenue earner [1].

One of the major sources of loss to the industry is the fungal root rot armillaria (Figure 1.1), which has caused production loss of up to 25% in first rotation plantations on indigenous forest sites [52]. Loss for second rotation stands are harder to obtain, as continually improving forest management techniques obscure potential returns from immature forests, but an estimate of 5-10% production loss seems reasonable [41, 35].

The only effective control measure for the disease in areas of serious infestation is the up-rooting and burning of previous rotation root structures before re-planting. Such labour intensive treatment is expensive, and cheaper alternatives are needed. Chemicals are generally ineffective against the fungus [20], which is well protected by the soil. Biological control may be the cheapest and simplest approach, as long as a suitable agent can be found. Prediction of the long term success of tree root biological control programmes is severely hampered by the long time scales involved, and the physical inaccessibility of the system. Field trials of potential armillaria treatments could be expected to take at least 8 years, this being the age by which most trees are largely resistant. The excavation and culturing required to monitor the disease's progress is very labour intensive.

Computer simulation avoids time and experimentation constraints. Even if simulation run times were several days, feedback is rapid compared to field trials. Simulation also naturally allows complex quantitative measurements of the modelled system to be made without any significant time / resource input. Providing no major simplifications are made, and acceptable validation is achieved, computer simulation of the geometry and development of the root system and associated microorganisms offers a technique for rapid and inexpensive assessment of possible control and management schemes.

For example, the interactions between two of the basic crop management decisions, inter-tree spacing and the thinning programme, have a complex effect on armillaria dynamics. Planting density is often higher than the desired final crop density, to compensate for various forms of mortality, but decreasing the inter-tree spacing may increase the rate of armillaria spread. Similarly thinning may either check armillaria by creating gaps between remaining trees, or increase armillaria impact by providing large volumes of readily available substrate. A computer model, run with various initial stocking densities and thinning schedules, offers a quick and cost effective way to balance these effects.

1.1 Increasing stand productivity

A versatile model of plant root architecture and ecology has a wide range of possible applications. In the case of the *Pinus radiata* armillaria pathosystem, the potential increase in monetary profit from more efficient root rot control could be taken as justification in itself. Alternatively, an environmentally based rationale may be used; an example of such was outlined by Libby [38]. Taking the proposed preservation of 6000 ha of productive Californian forest as an example, he examined the flow on effects of reducing supply in one area increasing demand in another. Under a sustainable management programme, the Californian forest's production would drop from 11 m^3 ha^{-1} to 5 m^3 ha^{-1}. This shortfall would be countered by increased imports from neighboring states, which (also under pressure to conserve) would consequently reduce exports to the far east, in particular Japan, who would in turn increase imports from the Philippines, Madagascar, Malaysia and other developing countries with rainforest reserves.

Using unavoidably questionable but generally conservative figures, Libby estimated that the loss of the 18 000 ha of rainforest required to cover the Californian conservation initiative could cost the extinction of some 90 species (across all ecosystem levels including microbes). That 18 000 ha of rain forest are required to match the production of only 6 000 ha of temperate forest is a reflection of the generally poor timber quality of species found in virgin rain forest. If the Californian forest is protected completely, the figure becomes 160 species. Clearly flow on effects need to be considered when conservation initiatives are promoted. One recent Tasmanian preservation action included a NZ $25 000 000 mitigation cost to increase the productivity of other areas.

Using Libby's figures, increasing the productivity of a *Pinus radiata* plantation by 1 $m^3 ha^{-1} yr^{-1}$ conserves 1 species for every 420 ha the increase is applied to. New Zealand has some 1.3×10^6 ha of plantation forest; the potential for reducing demands on global native forest ecosystems by increasing the productivity of production forestry is clearly considerable. A practical, economically and ecologically sound control strategy for armillaria, a major source of lost production in New Zealand's plantations, would have wide ranging benefits. A simulation model of the pathosystem could play an important role in the development of such a strategy.

1.2 The case study system

1.2.1 *Pinus radiata* in New Zealand

Timber has always been one of New Zealand's assets, from the early felling of the native forests to the present. Although some suggest other timbers should be used in many situations [55], *Pinus radiata*'s rapid volume increase and short generation time have made it the dominant species in commercial forestry [63]. Forestry is a major export earner for New Zealand, currently export earnings in the forestry sector are increasing, and more efficient forest management will probably sustain this trend [1, 35, 2].

1.2.2 The root system of *P. radiata*

Many authors have studied the root systems of conifers. Lutz, Ely and Little [40] provided a comprehensive white pine-root density study, combined with extensive soil profile data. However, as is typical of most subsequent authors' research, their measurements are mostly trench wall root intercept frequencies, and are more suited to root biomass estimation than root architecture prediction. Only when the trench passes directly through the centre of the root system do these studies give an indication of overall structure, and even then no allowance is made for the asymmetry generally present in the rooting patterns of mature trees. A detailed root architecture model is important when considering diseases like armillaria which are spread by direct root contact.

There are several contemporary studies for *Pinus radiata* in New Zealand. Santantonio and Santantonio [49, 50] examined the effect of thinning on fine root production by *Pinus radiata*. Data

were collected on biomass of fine roots at different depths. The plant's use of photosynthate for fine root production that might otherwise be used for increasing log volume is often examined, and is often found to represent a significant proportion of a tree's production. In these studies, on a healthy site, fine root production represented 4 to 6 percent of total production, which is not considered excessive. Where fine root turnover is unusually rapid, due to disease, grazing by soil fauna, or poor soil moisture conditions (flooding or drought), the loss of potential wood production may be much greater, with figures as high as 67 percent reported [49].

Working on biomass partitioning between below and above ground components, Jackson and Chittenden [33] measured root parameters of *Pinus radiata* in an attempt to find strong correlations with above ground parameters. Diameter at Breast Height (D.B.H., 1.4m above the ground) is the most common productivity measurement in the field. They found oven dry weight of roots to be related to D.B.H. as follows

$$\log_e(\text{root weight (kg)}) = 2.73 \log_e(\text{D.B.H. (cm)}) - 5.009 \qquad (1.1)$$

A result from their work which may be of considerable use to the validation of a root geometry simulation is the relationship between mass of a root system component and its length. For example, "large root" length in cm was determined to be $1.7 \times (\text{dry mass})^{0.991}$, and "fine root" length in cm was determined to be $851.2 \times (\text{dry mass})^{0.57}$, where dry mass is measured in grams after oven drying. In its inverse form this relationship could be used to convert root geometry estimates to more common biomass data. However there is probably a considerable element of site specificity in the parameters of equation 1.1.

Another common form of root study is the subjective scoring of a root system for attributes such as twisting and evenness of distribution. Balneaves and Mare [3] examined these parameters as a function of planting technique. They found that careless planting can lead to a weak, bent tap root, and a lopsided distribution of horizontal lateral roots that increase the danger of wind-throw.

Somerville [53] also examined rooting as a cause of wind-throw. Quantitative results were produced, essentially looking at root types by depth by treatment. Root types were classified as laterals (surface oriented, long, shallow roots), sinkers (vertical roots below the stump and branching from laterals) and obliques (roots falling somewhere between the other two classes). The treatments were various combinations of shallow and deep ripping. Naturally regenerated

trees were found to have the most substantial root systems, and the most resistance to wind throw. Root morphology was characterised by dividing the root volume into eight (as opposed to the more usual four) sectors, plus a central "core" cylinder. Some measurement, preferably quantitative, of root distribution is required if simulated root architectures are to be compared with those in the field.

O'Loughlin and Watson [43] also examined the strength of root systems, more specifically the decrease in tensile strength of roots of various diameters after felling, which determines the contribution of roots to slope stability of cleared land. Of great significance to the study of *Armillaria spp.*[1] (species of the genus *Armillaria*) distribution in newly planted areas are their measurements of decay rates for various diameter classes of root. Roots below 30 mm in diameter were found to decay progressively over 30 to 40 months, although extensive rotting at 14 months may have signalled the displacement of parasitic species like *Armillaria spp.* by more efficient, specialist decomposer species. The stump and larger roots may decay over 25 years or more, whether or not they remain viable sources of armillaria is unclear, but given its ability to live as a saprophyte, some low level persistence is possible.

In a study unusual for its precise geometrical data set, Watson and O'Loughlin [62] used sluicing techniques to excavate intact root systems of *Pinus radiata* after 8, 16 and 25 years of growth. Detailed drawings were made in plan and elevation of the undisturbed root systems, and these were used for determining distributions for branching frequency, angles and other parameters used in this research.

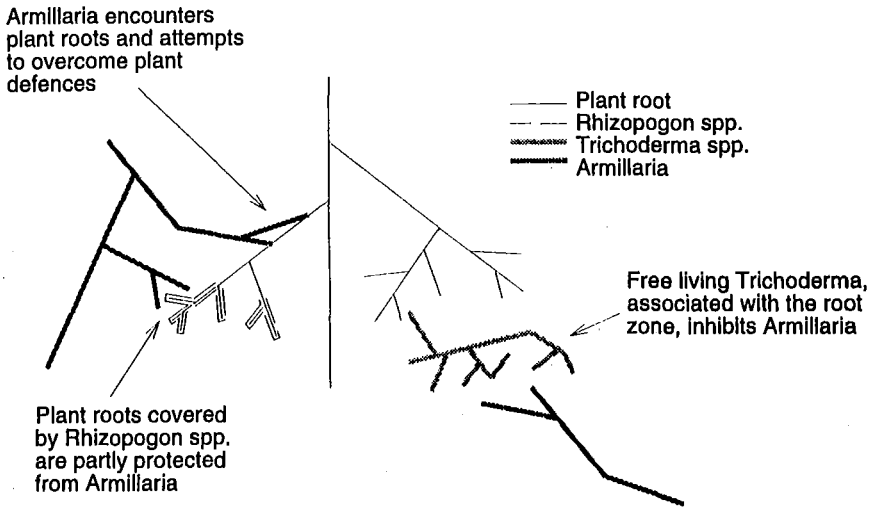
While New Zealand studies invariably examine *Pinus radiata*, other research deals with a variety of species. Carlson et al. [9] looked at surface (running parallel to the ground) and deep (approximately vertical) oriented roots in Loblolly pine. Severing treatments were applied to both groups of roots to assess their relative importance in nutrient and water supply. Severing first order horizontal lateral roots had the greatest impact, probably because this also removes vertical second order sinkers, and greatly reduces the soil volume exploited by the root system. The division between horizontal and vertical roots is a characteristic feature of many root systems and may form a useful basis for comparing model and field observations.

[1] The form *Genus spp.* indicates either members (species) of the genus in general, or a specific, but unidentified member of the genus.

A key root system study in geometric terms was performed by Henderson et al. [28]. Branching and bending information was extracted from detailed records of the positions of all roots for excavated Sitka spruce trees. This information was obtained specifically for a root geometry simulation [27]. A modified form of Henderson et al.'s work, with parameters derived from Watson and O'Loughlin, has been used for this work.

1.3 The organisms and their relationships

Even when taking into account only a small group of organisms, the number of potential interactions is considerable (Figure 1.1). The variation in modes of interaction (spatial exclusion, resource competition, antibiotic production, direct antagonism, predation), combined with the possibility of higher order interactions, makes evaluation of the relative importance of each organism difficult.



Fungus	Effect on *Pinus radiata*
Armillaria spp.	A root rot disease, responsible for tree mortality, particularly during early development, and reduced reduction in mature trees
Rhizopogon spp.	Mycorrhizal fungi that form symbiotic relationships with many conifers, including *Pinus radiata.* Known to increase plant survival and production.
Trichoderma spp.	An antagonist of other soil fungi, known to strongly inhibit *Armillaria spp.*, and thought not to adversely affect *Rhizopogon spp.* May also have some beneficial effect on seedling development.

Figure 1.1: The *Pinus radiata* fungal ecosystem.

1.3.1 *Armillaria spp.* as a pathogen of *P. radiata*

Armillaria kills and stunts young (0-8 years) trees by attacking structural roots and the base of the trunk, cutting off the root system from the rest of the tree. Direct contact between healthy roots and diseased roots, often left over from the previous rotation, is considered to be a major path of disease spread. Fungal rhizomorphs[2] also carry the fungus between neighbouring roots and trees. The timing of contact between healthy and diseased roots, and hence the positions of all roots in the soil, are important in determining the impact of the disease. Trees over eight years of age are largely resistant [30].

Hartig [26] gave a detailed account of disease in conifers caused by *Agaricus (Armalaria* [sic])

[2]A rhizomorph is an exploratory colonising structure produced by soil fungi. It is a narrow, root like hyphal organ, typically with little or no branching. It may be 3–4 metres in length.

melleus. Root rots and resin flux disorders caused by the fungus had previously been attributed to crowding in closely planted seedlings. The association of these disorders with tree spacing may be an indication of the importance of inter-root and inter-tree proliferation of the disease. *Armillaria mellea* is often used as a general term for a variety of *Armillaria* species, in New Zealand the common species are *A. limonea* and *A. novae-zelandiae*, the latter being the most virulent pathogen [52].

James et al. [34] estimated that root disease centres (areas of total tree mortality) occupied 1% of commercial forest land in seven conifer forests examined in North America. About 13% of commercial land had some root disease mortality, and 35% of total annual tree mortality was attributed to root disease. The major root pathogen was *Armillaria mellea*. More directly applicable figures were given by MacKenzie [41], who estimated volume loss to *Armillaria spp.* in a wide-spacing, 28 year rotation *Pinus radiata* stand in New Zealand at 6-13%. Close spacings could be subject to even higher losses.

Patton and Riker [44] induced death in several 7-10 year old conifer species by inoculation with *Armillaria mellea* grown on sterilised root segments. The segments were buried in the soil surrounding the tree roots, and infection was by rhizomorph growth. Shaw et al. [11] achieved similar results with *A. novae-zelandiae* and *A. limonea* inocula and *Pinus radiata*.

Armillaria spp. root rot was found to be the most damaging disease of *Pinus radiata* planted on recently cleared indigenous forest sites in New Zealand [52]. There was a direct relationship between percentage loss and previous indigenous cover, with a worst case of 27% fatality after 2 years on a site previously occupied by *Beilschmiedia tawa* (Tawa). Assuming a 60% reduction could be made in disease impact by stump removal, they found an expenditure of up to NZ $\$150\,\mathrm{ha}^{-1}$ (at contemporary prices) could be justified.

MacKenzie and Shaw [42] found a distinct clumping of *Armillaria spp.* induced mortality around stumps of *Beilschmiedia tawa*. They suggested a time dependent radial expansion of infection from these centres. Later work by Roth et al. [48] indicated a fixed area of infection with an expanding zone of maximum infection rate giving the illusion of disease spread. Pas [61] was not able to confirm this, instead a random pattern of mortality was proposed for disease patches centred on *B. tawa* stumps. It was observed that expanding root systems of new trees would eventually make contact, possibly giving rise to a radial expansion of diseased area, but

that the impact of this would be reduced by resistance to *Armillaria spp.* increasing with plant age.

All of the above-mentioned studies were for irregularly spaced stumps of indigenous trees in first rotation pine plantations. Armillaria's importance in second rotation forests is not yet as clear, but while it is generally considered to have slightly less impact than in first rotation stands, it remains a major source of plant mortality [30].

1.3.2 *Trichoderma spp.* as a biological control agent for *Armillaria spp.*

Fedorov and Bobko [21] found that *Peniophora gigantea* and *Pleurotus ostreatus* had potential for biocontrol of *Armillaria mellea*, and spread well in pine stands. Mycelium was applied to cracks in the bark. *P. gigantea* and *P. ostreatus* are both wood decay fungi, and are able to "starve out" armillaria by consuming available dead wood more efficiently, while not attacking healthy trees. Fedorov and Bobko [20] also found 6 fungicides tested were ineffective, as the mycelia and rhizomorphs of *Armillaria mellea* were inaccessible to the chemicals. These results suggest a fungal biological-control agent may be one of the only viable options for *Armillaria spp.* control.

More recent and conclusive work in New Zealand has isolated a specific anti-biotic compound, 6-pentyl-α-pyrone (6PAP), produced by *T. hamatum* and *T. viride* [17]. 6PAP has a strong suppressive effect on a range of other fungi, including *Armillaria novae-zealandiae.* The compound appears to trigger cell lysis. The trichoderma strains involved do not seem to be involved in direct parasitism; presumably trichoderma gains a general competitive advantage by suppressing other fungi. Initial trials with live trichoderma preparations in *Pinus radiata* plantations have shown a decrease from 22% to 6% armillaria induced mortality [17]. To scale this technology up to commercial plantations, and to obtain the maximum benefit, an ecosystem model operating at the individual root level would be useful in determining the placement of inocula and timing of application giving the maximum cost benefit.

1.3.3 *Rhizopogon spp.* as a symbiont of *P. radiata*

The root structures of *Pinus spp.* and the modifications made to them by mycorrhizal fungi have been studied extensively. The distribution of mycorrhizal fungi in long roots of *P. sylvestris*, as indicated by the presence of Hartig net structures, was recorded by Robertson [46]. Even in the absence of the familiar forked lateral roots, mycorrhizal hyphae were found in long root tissue, so it seems unlikely that mycorrhiza free root systems occur with any significant frequency.

R. vinicolor applied to Douglas-Fir (*Pseudosuga menziesii*) was found to increase plantation survival and production after two years, although no difference could be observed in the nursery [10]. Although indigenous fungi colonised roots after planting out, *R. vinicolor* persisted in existing root structures and colonised those that developed subsequently.

The establishment of mycorrhizal populations in the pine-root system is a complex process. The dependence of plants on mycorrhizae, usually for phosphorus, is governed by the nutrient status of the soil. However, the microbial ecosystem in place also plays an important role. Garbaye and Bowen [23] found microflora of several soils to have both positive and negative (but more commonly positive) impacts on mycorrhizae development. Their choice of a sterile soil as the control measurement, although obvious, may have influenced the effects observed in favour of positive results. While autoclaving may have initially released nutrients into the soil, during the 4 month duration of the experiment the decreased soil species diversity may have decreased the amount and diversity of compounds available. It seems reasonable that a sterile soil, providing only very low levels of the various nutrients and co-factors found in a biologically active soil, would in itself be a negative effect on mycorrhizal development.

Assuming significant mycorrhizal development does occur, as is invariably the case in soils chosen for forest production, the pattern of development of the fungus over the root system is also significant, particularly when viewed in terms of forming a physical barrier to other fungi. Robertson [46] examined the pattern of development of various mycorrhizae over the surface of active *Pinus sylvestris* roots.

Mycorrhizal roots are characterised by "long roots" with "short roots" (often Y shaped branches) every few millimetres at their active ends (Figure 1.2). Mycorrhizal hyphae grow out into the soil volume to form foraging and reproductive structures. Wong et al. [64] gave figures for the

formation of mycorrhizal roots in *Pinus banksiana.* Fine roots were covered by 2–13 layers of fungal hyphae, at a density of 280–400 hyphae per millimetre, effectively forming a dense outer layer around the roots.

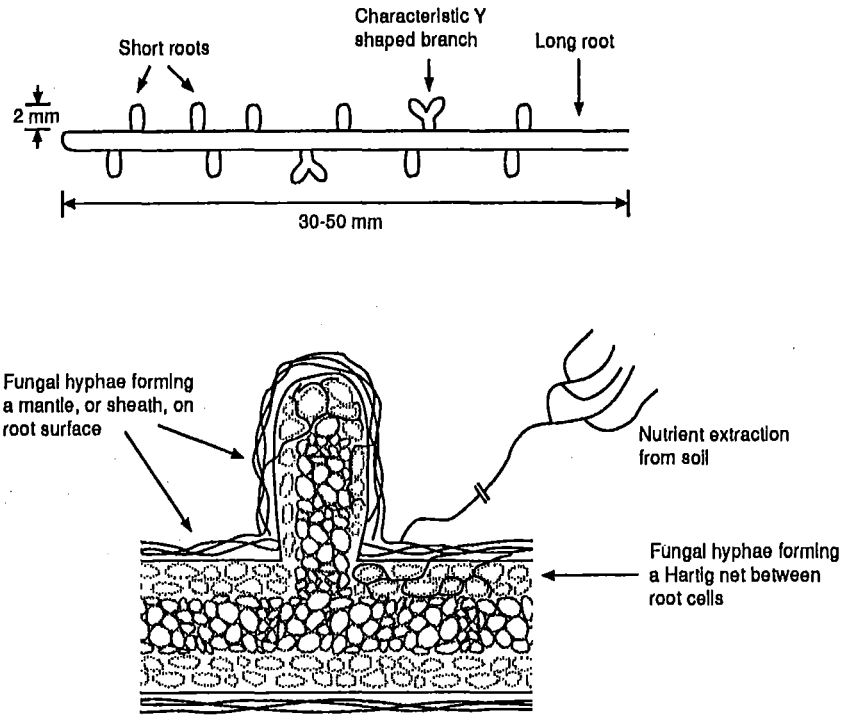


Figure 1.2: Mycorrhizal colonisation of *Pinus spp.* roots

Work on pinus mycorrhizae in New Zealand has concentrated more on distribution of species between geographic regions [13, 15, 16], although a limited amount of research on plant response showed *Rhizopogon spp.* to be the most effective in increasing the rate of *Pinus radiata* development [14].

1.3.4 Effect of *Trichoderma spp.* on *P. radiata.*

Although the relationship between non-mycorrhizal, non-antagonistic, rhizosphere fungi and plants is often not well characterised, there is some evidence to suggest *Trichoderma spp.* may stimulate *P. radiata* development in the absence of armillaria.. This could simply be an indication of suppression of some other, undetected fungal antagonist of *P. radiata*, although a more direct effect may exist.

1.3.5 Interaction of *Trichoderma spp.* with *Rhizopogon spp.*

The relationship between *Trichoderma spp.* and *Rhizopogon spp.* has not been studied extensively, as *Trichoderma spp.* are normally applied as a biological control agent (BCA), and *Rhizopogon spp.*, being beneficial symbionts, have not been the subject of biological control programmes. Recent trials in New Zealand have indicated that there is no significant adverse interaction [29].

1.3.6 Interaction of *Rhizopogon spp.* with *Armillaria spp.*

Although *Rhizopogon spp.* are unlikely to react directly to *Armillaria spp.*, they may play an important role. As specialist mycorrhizal fungi, they colonise their host without eliciting any significant host resistance response, which gives them an advantage over any fungus that must overcome host defence.

As ectomycorrhizal fungi, *Rhizopogon spp.* form a potentially quite dense sheath over the active roots of the plant, and they often decrease the numbers of exposed "fine" roots the plant produces, by replacing the fine roots' function. As active and fine roots are generally the most susceptible to attack, *Rhizopogon spp.* may reduce *Armillaria spp.* infection simply by physical exclusion. However, this form of protection may be more relevant with vascular wilts, or bacterial diseases; Stanosz and Patton [54] found rhizomorph penetration in heavy structural roots of aspen suckers; such roots are non-mycorrhizal.

1.4 Current status of the armillaria pathosystem

The following are some of the questions raised by Hood [30] concerning armillaria root rot management:

1. How does inoculum enter and develop in a stand?
2. Does chronic infection originate predominantly from previous crop inoculum, from new inoculum introduced at the beginning of the rotation period, or from inoculum entering through thinning stumps?

3. How important is pine to pine spread of infection?
4. How intensively, and when, should a severely infected stand be thinned to avoid losing final-crop trees?
5. Does thinning increase the level of chronic disease by producing new inoculum through spore formation, and by furnishing existing inoculum with an additional food substrate?
6. If so, is stump size, and therefore timing of thinning important?
7. Or does thinning ensure greater vigour and perhaps improved resistance for final crop trees (not forgetting that more vigorous root systems may enhance the chances of chronic infection through increased root contact with inoculum – (Shaw et al. [11])?
8. What is the effect of pruning?
9. When should infected stands be harvested?
10. What is the effect of current management on the level of Armillaria in future stands?
11. Will the disease continue to build up on new sites, with more extensive early mortality?
12. How widespread are stands with chronic infection, and how evenly distributed is within-stand infection?
13. When in the rotation is the best time to apply control or management measures?

As an increasing number of exotic forests mature, and second rotation forests are becoming more numerous, *Armillaria spp.* continues to cause high rates of mortality in young trees, and hence to be a major factor in stocking density decisions.

1.5 The root system structure

The modelling of the growth pattern of the root system is a major step in the overall simulation. Although a plant root system may appear largely unstructured, root growth can be considered as a structured process, whose random appearance can be attributed largely to its typically heterogeneous soil environment [27]. All root systems can be viewed as a hierarchy of different orders of roots, where the first order roots originate from the base of the trunk and branch

to give rise to second order roots which branch to give rise to third order roots (Figure 1.3). Figure 1.4 shows a 2 year *Pinus radiata* root system excavated without displacement, exhibiting the characteristic division between horizontal and vertical roots. In Figure 1.5 the root system has been pulled from the soil by a winch. Longer roots of less than 5 mm diameter are missing, but the overall structure is apparent. The background of these trees is outlined in section 4.3.

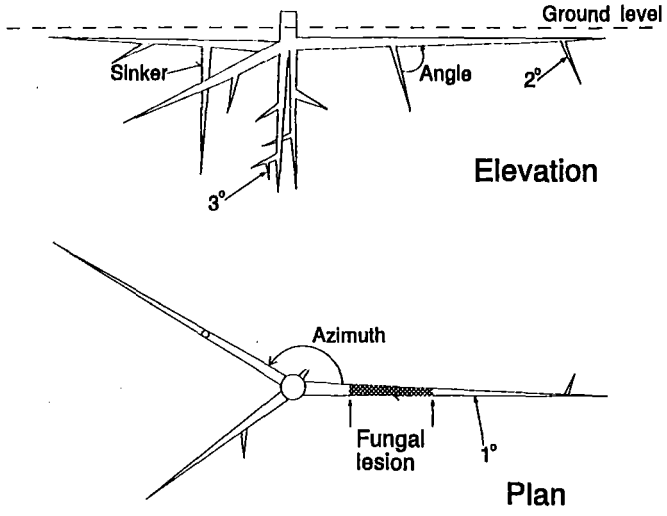


Figure 1.3: Root system structure.

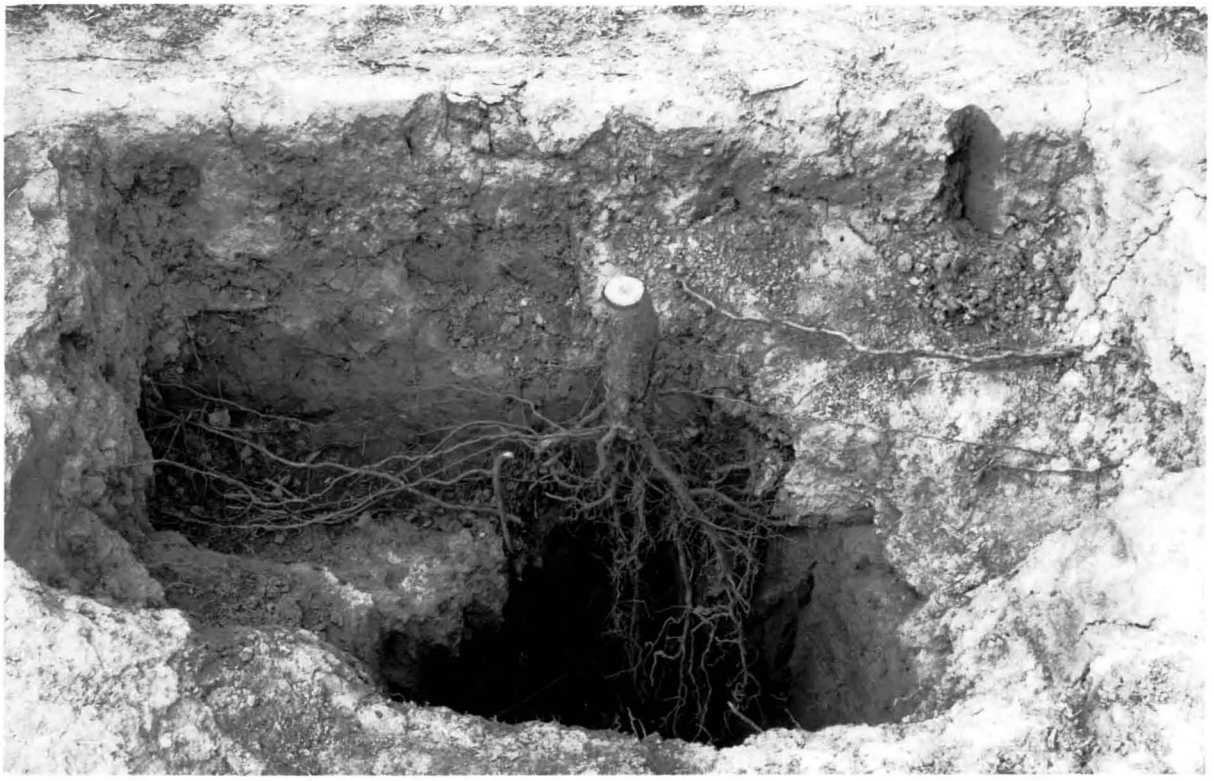


Figure 1.4: Two year old *Pinus radiata* root system *in situ*.

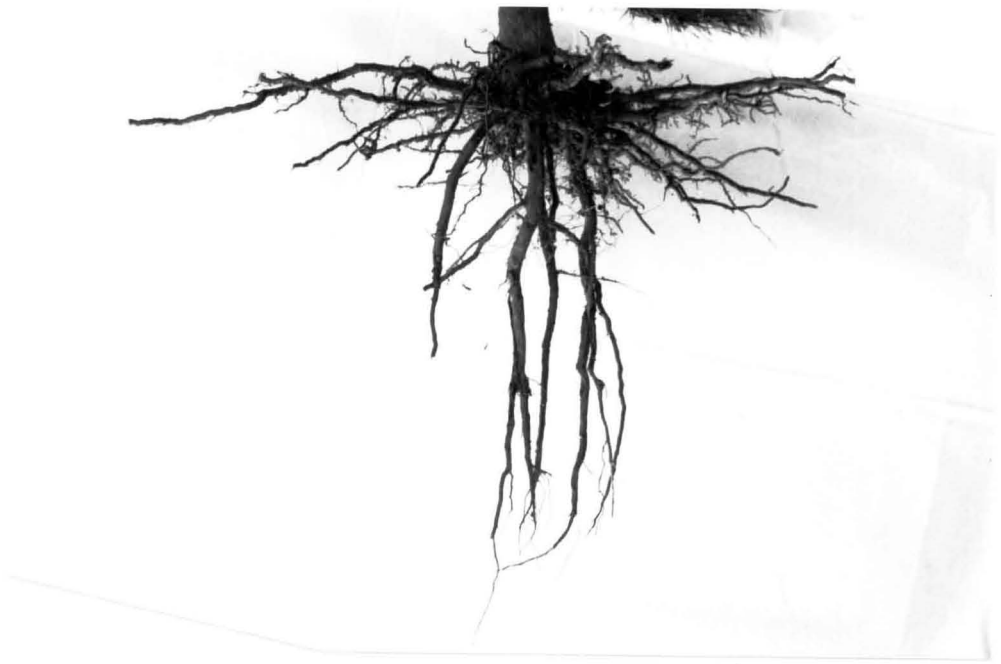


Figure 1.5: 2 year old *Pinus radiata* root system.

Typically first order (1^o) roots will be thick woody structures and fulfil a structural support role as well as facilitating nutrient transport. Higher order roots, particularly third order and above, may be more physiologically active and involved in nutrient uptake. *Pinus radiata* first order roots may be divided into vertically oriented roots directly below the trunk, which are involved in water uptake and support, and horizontally oriented roots that distribute the higher order root system components over a larger area for nutrient uptake. Vertically oriented second order roots, called sinkers, may also be present.

Recent research and review of the armillaria problem has highlighted many unanswered questions concerning effective management practices. Hood's [30] summary found the effects of many forest management practices (thinning, spacing, planting time) on armillaria were largely unknown. A computer simulation of the growth of seedling root systems would be a signif-

icant step towards a better understanding of this important disease. Such a simulation would encompass seedling root system development, population dynamics of relevant fungi, and the process of decay in the root system and stumps of the previous crop, which is believed to be the primary source of armillaria infection. A computer model allows many scenarios to be evaluated in a fraction of the time required for field trials. A fully configured model may be able to answer in one or two weeks questions that would require field trials running for many years. Even if parameterisation time is required, model results should be available in months rather than years. Of course, there will always be problems models cannot answer, usually because of poorly characterised mechanisms, and model output can only be used with confidence if solid validation results are achieved.

1.6 Objectives

The objectives of this research are given below:

1. To develop a computer simulation of spatial distribution of root by class (root hairs, fine roots, structural roots, mycorrhizae) for *Pinus radiata* seedlings.
2. To validate the model using available field data.
3. To define the space resource available to members of the microbial community, using the simulation.
4. To simulate the development and distribution of *Armillaria spp.*, *Trichoderma spp.*, and *Rhizopogon spp.* population in the soil surrounding *Pinus radiata* seedlings.
5. To validate the simulation using existing stand loss and disease distribution data.
6. To examine optimal strategies for seedling production and establishment in terms of these fungi, using the simulation model.

Chapter 2 lists the features required in a model of the *Pinus radiata* / armillaria pathosystem, and examines some existing models of plant root structure. Chapter 3 then develops the structure of the simulation model presented here. Chapters 4, 5 and 6 consider the collection of the required parameters, some implementation details, and possible approaches to validation. In chapter 7 the simulation's responses to thinning, disease spread and biological control scenarios are examined. Chapters 8 and 9 look at the simulation's performance and its possible future development and application.

Chapter 2

Model requirements and Existing Models

2.1 Flexibility

While the work described here was prompted by questions raised by the *Pinus radiata* / armillaria pathosystem (section 1.4), the simulation problem, the representation of a multiple order root architecture and the fungal populations associated with it, is common to most root disease and root ecology systems. Hence any modelling system developed to examine a specific pathosystem from the root architecture perspective should ideally be easily reconfigured to simulate other plant and fungal species. Such flexibility suggests the need for the description of the root architecture to be isolated from the rest of the simulation's implementation.

The overall objective of this work is not so much to answer specific questions about a specific root ecosystem, but to develop a simulation modelling system that can represent some of the diverse structures and relationships found in root ecosystems. Specific examples of the ecosystem components are examined in the following section. There are three levels at which flexibility may be considered. Firstly there are straight forward parameterisation values such as elongation rates and lifespans. Secondly there are variations in the "behaviour" of some parts of the ecosystem, for example some root systems are insensitive to their position relative to the stem, while others may branch more profusely within the area directly below the foliage. In such a case, when configuring a model it may sometimes be sufficient to define second order roots as branching on average 4.8 times, while in another case second order roots may branch on average

4.8 times if the root in question is within 1.2 m of the stem, or 2.1 times otherwise. Thirdly there is the ease with which a model may be adapted to address problems other than those for which it was originally designed. For example, some fungal pathosystems include mobile soil invertebrates as important disease vectors. The ability to model the movement of these entities through the simulated system may be important in such cases.

The first level of flexibility, the adjustment of simple numeric parameters, is required in almost all models, and poses no implementation problems. The second level, parameter dependence on other system elements, is more difficult to achieve if the simulation is to remain reasonably simple and accessible to a wide range of users. However, increasing complexity to increase capability is usually justified, as long as the rate of return (of capability for complexity) is satisfactory. The third level, completely new fields of model application, will usually require a significant degree of familiarity with a model's implementation. Even so, a model intended for wide ranging applications should be designed to allow the addition of new features without major revision of the basic internal structure.

2.2 Investigation of diverse environmental factors

2.2.1 Root Architecture

Root architecture is generally an independent variable, or model input, in the armillaria pathosystem (although armillaria induced root death can modify root distribution). However, the complexity of root structure means that its representation is a major part of the modelling exercise. The rate of armillaria spread between trees, and the rate at which seedlings encounter diseased material (stumps etc.), is governed by the number, length and growth rate of first order lateral roots in particular. The persistence of armillaria in the soil and the distribution of rhizomorph initiation sites is determined by the spatial arrangement of the thicker, more persistent, lower order sections of the root system.

2.2.2 Root physiological status

Any detailed model of root-fungi interaction must take into account the physiological state of the roots. This effectively imposes an "age structure" on the root population, in addition to

the differences between root orders. As with any population model for individuals with age dependent behaviour, the influence of random and historic events on the distribution of age groups forces the introduction of at least some simulation, as opposed to purely mathematical, techniques to the model. In this case, root growth has been divided into the following stages: Pre-emergent, Growing, Mature, Senescent, Dead, and Decayed. The effects of physiological age are discussed in section 3.5.

2.2.3 Fungal populations

Unlike above ground plant disease models, where spore mobility and foliage homogeneity allow fungi and hosts to be modelled by 1-dimensional population values, realistic modelling of below ground microbial populations often requires that the spatial component be taken into account. Many micro-organisms are able to enter states of near stasis in unfavourable locations, so the existence of viable material in cultured samples may not indicate that a species is active in the ecosystem. The dependence of microbial population density on distance from the root surface and the limited dispersion rate of infective material invalidates many simple growth equations. Even with excess nutrient availability, rhizosphere fungi are restricted, to a greater or lesser degree, to growth along the root surface, and so cannot achieve exponential population increase. To further complicate the population dynamics, the limiting space resource fluctuates in an essentially stochastic, step-wise manner, as expanding fungal populations encounter uncolonised roots, either because of the close approach of one root to another, or the action of exploratory hyphae.

In the armillaria pathosystem test case, dedicated mycorrhizal *Rhizopogon spp.* are closely associated with the root surface, although nutrient scavenging hyphae increase the chance of locating new roots. Generally mycorrhizae grow with the host plant; the tightly integrated nature of the symbiosis essentially creates a new, hybrid, organism. Weakly saprophytic armillaria is less rigidly restricted to the root surface zone, but it is still dependent on access to roots for much of its life cycle. The rhizomorphs armillaria produces are a specialised, although random, resource location device. Finally, *Trichoderma spp.* are true soil fungi, with no direct bond to the root surface. Clearly at least three distinct population dynamics must be modelled for this pathosystem.

2.2.4 Armillaria induced mortality

The impact of armillaria infection is dependent on the age of the tree affected. Most mortality occurs within the first 8 years, and trees over 15 years old are generally at risk only when armillaria inocula levels are high and environmental conditions favour the fungus. Mild infections of only a small proportion of a tree's roots may retard growth and lead to uneven stand development, which detracts from the economic value of the harvest. Most production forests are planted at higher densities than that intended for their final, mature harvest. The initial density often takes into account expected rates of tree loss to disease, as well as planned thinning events. But because armillaria spreads by inter-tree contact, adjusting the initial planting density alters the dynamics of the disease for which the adjustment was intended to compensate. For example, increasing the stocking density to allow for an expected loss rate may result in an even greater loss rate, as inter-tree root contact is increased.

2.2.5 Root and fungal interaction

The obvious prerequisite to interaction between fungal populations is physical contact, or at least close proximity. This can be modelled on a probability basis, but the outcome of the interaction depends on more than simply whether or not contact occurs. In the case of armillaria, the success of an attack on a root depends on size of and distance to the resource base of the attacking fungus [24]. A rhizomorph or inter-root contact originating from a nearby site of extensive armillaria infection is more likely to succeed than one originating from a distant lesion or small amount of substrate.

Similarly the outcome of contact between disease and biocontrol species depends on the vigour of the two hyphal populations. The method chosen for biocontrol introduction to the ecosystem may also complicate contact probability. As a hypothetical example, consider a biocontrol treatment known to reduce overall armillaria inocula levels by 70%. If the critical inocula level, at which the plant is able to resist the pathogen, is 50% of the initial level, then it is possible that what looks like a reduction to a sub-critical level may not in fact represent a successfully controlled situation when alternative spatial distributions of the reduction are considered (Figure 2.1). This suggests a need to consider the spatial arrangement of the root system.

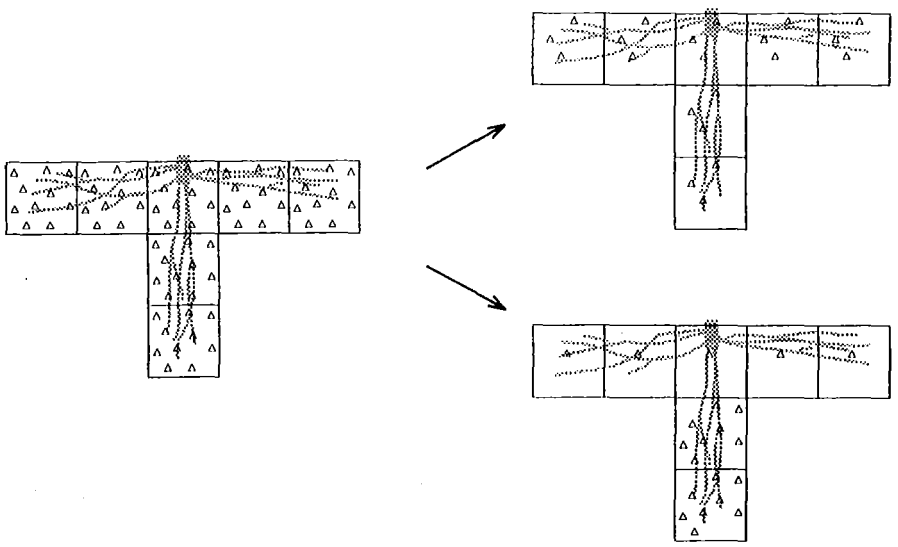


Figure 2.1: Spatial distribution of inocula reduction, in both cases the overall reduction is 70%.

2.2.6 Thinning

The sudden introduction of dead roots to the system has a marked effect on disease dynamics. Some pathogens, like the moderately saprophytic armillaria, are favoured by the sudden decrease in the roots' resistance to attack. However, it does not follow with certainty that thinning will worsen armillaria infection; some studies suggest the rapid consumption and growth that occurs is less damaging overall than a prolonged, lower level of activity. Other pathogens, less able to compete with dedicated saprophytes, may be rapidly excluded from the dead substrate.

2.2.7 Ground topography

The soil volume within which a tree root system exists is not necessarily bounded by a horizontal plane at ground level. The tree may be growing on a sloping hillside, or, in many production forestry situations, on or between mounds and furrows of soil and slash material created by the harvesting of the previous crop. Therefore, a spatially oriented simulation of tree root architecture should be able to represent the shape of the ground surface. Henderson et al. [27] modelled root architecture in raised mounds, and adjusted a root's direction of growth to run parallel to the ground surface, whenever a root would otherwise have passed through the surface.

2.3 Model requirements – Summary

To satisfactorily model root form and function, fungal populations, plant / fungal interaction, and the effects of forest management and locality, the simulation developed must be able to model root system morphology and physiology, microbial populations' state and location, the interaction of plants and microbes, and the impact on the system of discrete, possibly random events, such as root contacts, thinning and biocontrol applications.

2.4 Frameworks for spatial models of root ecosystems

There a two possible frameworks for spatial models of root ecosystems; either simulation of root growth physiology, or characterisation of root system morphology. There are also two ways to implement a spatial model of a root ecosystem; either as a data structure defining the shape and position of each individual root, or as some density function that gives a probability of a certain type of root occurring in a certain volume of soil. Figure 2.2 illustrates the relationships between these approaches, and places some existing models in this framework. Note that the number of models placed in each category in no way reflects the relative importance or popularity of each category.

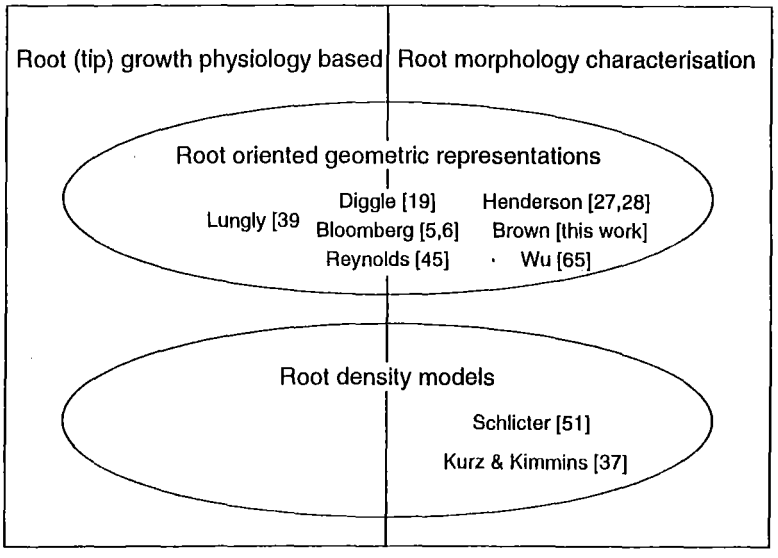


Figure 2.2: Frameworks for spatial models of root ecosystems.

2.4.1 Root growth physiology models

The physiology and control of root tip growth have been extensively studied. The elongation of the root tip is directed largely by chemical signals, although the exact course of the root through the soil is determined by the physical properties of the soil, such as bulk density and particle size. Tracking the progress and branching of root tips allows a three dimensional map of a root system to be created over the duration of a simulation. Lungley [39] used this approach to examine cereal development, and found that the interaction between elongation and branching patterns may allow a flexibility of plant response to soil treatments. Bloomberg [6] extended this technique to simulate the spread of a root rot fungus through the soil and roots in a Douglas Fir nursery.

Bloomberg's [5, 6] simulation model, which recorded the state of every root with respect to time, required the estimation, mostly by laboratory pot trials, of over 25 parameters for 10 different aspects of seedling growth such as germination rate and heatsum / growth relationships for the first 180 days of seedling development. Each 5 day iteration period added new roots to the system from germinating seeds, and extended existing first order roots. Second order roots were added at regular intervals along the first order roots. The positions of all roots within a three dimensional lattice were recorded, allowing the spread of disease from root to root and from spores randomly positioned in the lattice to be simulated. Final mortality rates agreed well with figures from the field; the model was within 10%–30% for a range of indices, including proportion of root length infected and number of infected seedlings / roots.

2.4.2 Root morphology characterisation

Measurements taken from excavated root systems have been used to produce computer generated root systems. Wu [65] used the roots diameter to predict its behaviour, while Henderson [28] included the number of primary roots, their length, the frequency of bends and branches along their length and similar parameters for higher order roots as primary variables.

Henderson's simulation took the form of a "geometric construction" of a Sitka spruce root system. For example:

Given n 1^o roots (where N^o indicates a root of order N, see Figure 1.1)

$$\theta_1 = \frac{2\pi i}{n} + R_1 \tag{2.1}$$

θ_1 = initial azimuth for 1^o roots, radians

R_1 = random variable from the normal distribution $N(0, 0.11)$

$$\phi_1 = R_2 \tag{2.2}$$

ϕ_1 = initial angle for 1^o roots, radians

R_2 = random variate from the uniform distribution $U(\frac{-\pi}{18}, \frac{5\pi}{18})$

$$L_1 = R_3 \tag{2.3}$$

L_1 = the length of a 1^o root, cm

R_3 = random variate from $U(0, 65)$

$$b = R_4 \tag{2.4}$$

b = branches per root

R_4 = random variate from the Poisson distribution, $\lambda = k_o * L_p$

k_o = constant for each order

L_p = parent root length, cm

$$l_i = \frac{iL_p}{b} + R_5 \ . \tag{2.5}$$

l_i = position of i'th branch, cm

R_5 = random variate from $N(0, 0.0067)$

The number and position of bends were controlled in the same way as the branches. Changes in the roots angle and azimuth at branch and bend points were determined by a similar collection of distributions and parameters.

A major feature of Henderson's work was the application of heuristic rules to the geometric construction. For example successive changes in azimuth at bends had alternating signs, insuring the root grew away from the trunk and did not loop back as a result of 2 or 3 turns to the same side. Wu's and Henderson's computer generated structures provide a representation of a root system at one moment in time, and do not require growth rate information.

2.4.3 Root Oriented Geometric Representations (ROGRs)

The root geometry models used by Lungley [39] Reynolds et al. [45], Henderson et al. [28, 27], and Diggle [19] allow almost any parameter to be incorporated in the model, but require considerable computing resource when the number of seedlings or root elements in the simulation is high. In these structural simulations, the position in three dimensional space of each root or segment of root is recorded. Roots can be regarded as lines, cylinders, or tapered cylinders, and their areas of influence and points of intersection calculated. Curving could be represented in a number of ways (arcs, splines, high numbers of line segments), but one or more distinct, instantaneous changes in direction are probably the most computationally straight forward, without sacrificing any significant degree of realism. Fungal mycelium can be viewed both as an attribute of the roots themselves (infected, ectomycorrhizal) and as distinct branching structures (exploratory hyphae, rhizomorphs). Additionally, stationary fungal inoculum, such as sclerotia and basidiospores, may be regarded as points or spheres, with which elongating roots can interact.

2.4.4 Root Density Models (RDMs)

The probability distribution approach is less computer-resource intensive, but requires more assumptions about the behaviour of organisms in the soil. A major advantage is the suitability of many existing data sets, usually collected for biomass studies, for parameter estimation in these models.

Any point in the soil below a seedling can be identified by an angle of declination and a distance from the stem, if the cross section is assumed to be uniform for all rotations. The existence of some prevailing direction of soil-water flow, or previous soil disturbances that confer some form of alignment to the soil structure, may weaken this assumption, but their effect is unlikely to be

major. This approach would allow the distribution of different classes of root to be described by functions of angle and distance, so the probability of encountering a root of a certain class at a specific point could be calculated.

However, development of the above mentioned functions is a major exercise in itself. The choice between geometry or density approach has been weighed carefully (Table 2.1). Initially it was considered possible that a combination of the two methods would be useful, with geometric models simulating root contact events, and probability functions modelling, for example, free soil fungal populations. In practice the level of realism imposed by the geometric sub-models discourages more superficial techniques, and once the basic framework for a spatial simulation has been laid, there is little reason not to use the approach for the entire system.

	Root oriented geometric representations	**Root density models**
Theoretical formulation	Clear cut mechanistic approach does not require too many assumptions	Considerable development required to ensure chosen distributions provide an accurate model of population interactions
Computer implementation	Relatively small set of well defined problems	Dependent on range and diversity of sub-models developed, although not necessarily more time consuming than the geometry approach
Simulation run time	Possibly significant, even allowing for efficient coding and reasonably fast hardware	Should not be excessive, allowing for more parameter manipulation.
Validation	A reasonable level of confidence should be possible	Individual parameters may have greater effects, which is not necessarily desirable
Flexibility	Very flexible	May be limited

Table 2.1: Comparison of geometric and density approaches.

2.5 Existing root growth simulations

Root growth simulations have been used to examine many aspects of plant production dependent on the root system. Lungley [39] considered the effects of delaying application of a fertiliser supplying a vital nutrient on cereal-crop root-system structure, and found that the relationship between branching and elongation may damp the effects of such treatments. Schlicter et al. [51] tested the sensitivity of water-uptake models (used to estimate total transpiration in a stand) to changes in fine root parameters to assess these models in view of the difficulties associated with making these measurements. They found that quantity and distribution of these roots played an important role in these models, and stressed the importance of realistic measurements. Kurz and Kimmins [37], who examined the problems associated with distinguishing live and dead fine roots, emphasized the importance of accurate figures for fine root dynamics. Simulating the effect of different sampling strategies for known fine root turnover rates, they found that failure to observe simultaneous mortality and production of fine roots could result in over or underestimation of fine root resource allocation by 15 percent.

2.5.1 Douglas Fir root architecture model for a single fungal species

Bloomberg [5, 6] and Reynolds et al. [45] simulated the spread of *Fusarium oxysporum* in Douglas and Fraser Fir nurseries by treating the roots of individual seedlings as groups of cylinders in three dimensional space. The axis of elongation of these simulated roots was tested for intersection with propagules positioned randomly in the lattice. The expansion of lesions on the root surfaces resulting from these intersections, and the subsequent root to root transfer of infection when roots intersected each other was used to estimate disease spread. Both studies achieved reasonable success, given the highly random nature of the system they simulated.

Data for the distribution and elongation rates for the root cylinders and for the extension rates of lesions on the root surface were collected from pot trials. Validation by comparison with field trials was reasonable allowing for unforeseen surface flooding in some trials [45].

2.5.2 Simulation of cereal root system architecture

Diggle [19] produced realistic 3-dimensional models of cereal-crop root systems by treating each root tip as a moving particle. The tips' paths through the soil were tracked, and used to build up lists of points describing the shape of each root. The approach readily allows the simulation of root growth response to such things as soil layers, soil particle size, and zones of higher or lower nutrient concentration. System characteristics that are only obvious at larger scales (lateral / vertical relationships, sinkers etc.) may be more difficult to produce, although a system incorporating different over-riding traits in the behaviour of different tips should cope with most patterns. The approach is probably more ideally suited to the intricate fine-root systems of smaller plants than to the largely structural layouts encountered in tree root systems.

2.5.3 Root density approach

Schlicter et al. [51] and Kurz and Kimmins [37] have used the type of data set shown in Figure 2.3. Note that only type (b) gives information for lateral spread of the root system, although type (a) may often be derived from a series of soil cores taken at increasing distances from the stem. Schlicter [51] used 5 spatial distribution patterns to examine effects on water uptake models, ranging from very rapid attenuation with depth, giving a shallow root system, to no attenuation, giving a hypothetical uniform cylindrical distribution. When water uptake potential was considered, the shape of the root system was shown to be most significant for low root densities.

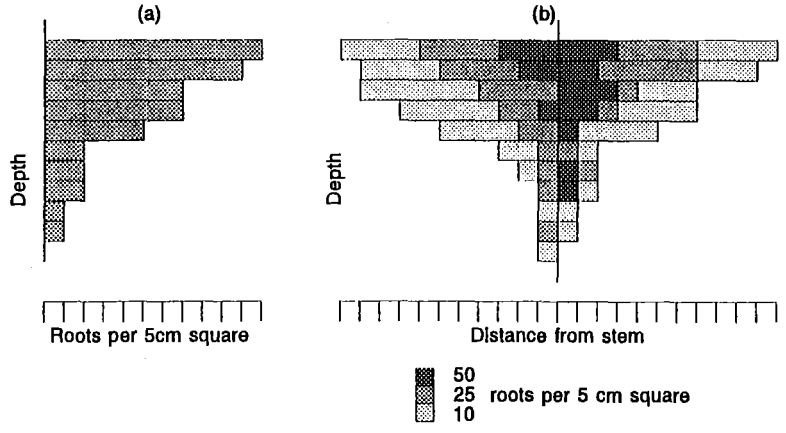


Figure 2.3: Root density data sets.

2.5.4 Length/branching ratios for canopy model

Charles-Edwards, Doley and Rimmington [12] define branching ratio and length ratio (σ_b and σ_l) for above ground plant structure:

$$\text{number of branches on a branch of order } O = N_b = \log_{10}(\sigma_b O) \tag{2.6}$$

$$\text{mean length of a branch of order } O = L_b = \log_{-10}(\sigma_l O) \tag{2.7}$$

However these equations were applied to above ground plant structure. The environment of the canopy is heterogeneous only in terms of prevailing wind, shading by adjacent trees, and sun angle. A reasonably regular and well balanced structure can be expected, and plantation trees are generally very similar above ground. But below ground, many factors cause variation between root systems. Initial planting may orient all the roots in a certain direction, an effect the tree never recovers from [57, 58, 59]; and rocks, compacted and loose soil pockets, other root systems, moisture gradients and pathogens contribute to an irregular root system structure.

2.5.5 Discrete, stochastic vs. continuous models

Stewart et al. [56] proposed the use of discrete event simulation for modelling of animal populations subject to a regulated hunting season. Stewart et al. considers a simulation tracking the life history of each animal and its interaction with hunters and other animals to provide a more flexible and reliable model of this relatively complex system. While traditional differential equation techniques provide realistic solutions for many simple systems, Stewart et al. suggests more complex systems governed by chance interactions between individuals, as opposed to continuous growth processes, are better represented by stochastic, discrete event simulations. Given that the outcome of disease / root encounters is determined by the history and status of the individuals involved, and that these encounters are infrequent, stochastic events, it seems reasonable that plant root / fungal interactions may also be modelled realistically by a simulation that models the development of individual roots and fungal populations.

Chapter 3

Model Development

3.1 Spatial envelope for temporal development

The physiological growth models of Lungley [39] and Bloomberg [5, 6] are probably the most rigorous method for root system simulation, but they require a large number of parameters to be estimated. Also, in the case of a study that covers the complete lifespan of the tree, not only the parameters, but also the underlying physiological models would need to be changed at various stages during the simulation. The geometric construction models of Wu [65] and Henderson [27] give a view of the root system at a specific stage of growth, but are not intended to model temporal development.

The technique developed here combines the two approaches of root growth simulation [5, 6, 39] and geometric construction [28, 27, 65]. The path of the expanding root system is defined by the pre-generated envelope created by geometric construction, and the extent to which the envelope has been filled, or the length to which the roots have grown, is defined by available growth rate data. Higher order sections of the root system complete their cycles of growth many times in the period required for the lower order roots to reach their full size. Fine roots may grow and die from a thicker parent root continuously throughout the parent root's lifespan.

The combined approach allows tree development to be simulated over a long period of time without constant revision of physiological sub-models, while still providing the temporal information necessary to predict microbial communities' impact on the tree.

The extension rate of the root is all that is required to model the root's growth, and the pre-defined envelope ensures the root's shape will be that predicted by the root architecture sub-model.

3.2 Root architecture – static components

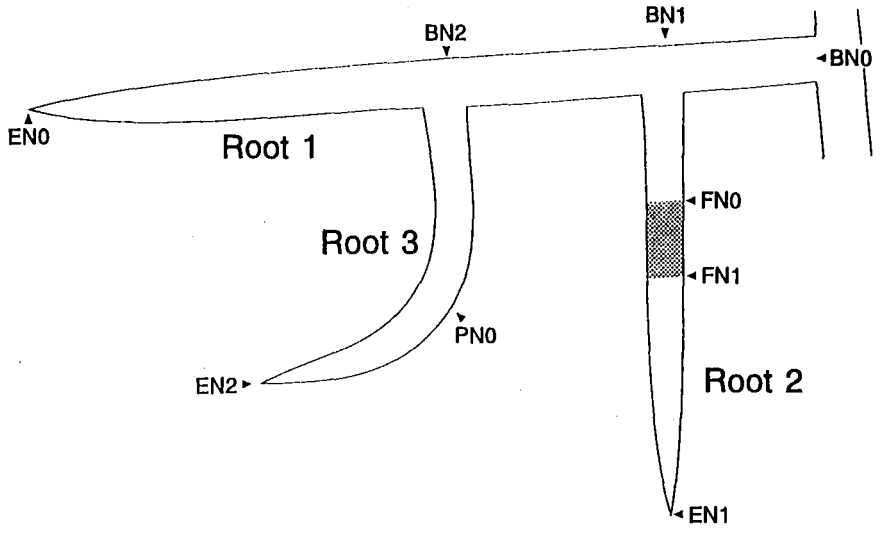
3.2.1 Nodes

A root system is considered to be set of nodes in a hierarchical list structure. Each node (or list entry) occurs somewhere along a simulated root, and has a position in 3-dimensional space (an x, y, z coordinate). A node may indicate branching, bending, or a change in a root's fungal status. All "features" of a root must be represented by a node. The internode, therefore, is a homogeneous length of root running from one node to the next. The internode's order and location are defined by the sum effect of all the nodes occurring before the internode, tracing the lists of nodes from the first ("root collar") node.

The types of node defined in the model are:

1. A "Bend" node is simply a point in space through which the root passes.
2. A "New Root" (or branch) node marks a new root header, the beginning of a new root of the next higher order.
3. A "Fungi" node represents a change in the fungal status (mycorrhizal, infected) of the root. A pair of such nodes could define a fungal lesion covering a certain length of the root. Moving the nodes up and down the root would allow the lesion to expand or contract. When the mobile nodes passed "Bend" and "New Root" nodes their position in the node list would change. Passing a "New Root" node would indicate the possibility of an infection spreading onto that new root.
4. A "Root End" node defines the position of the end of the root when fully grown.
5. A "Current End" node, which is invariably positioned on a line between two other nodes, indicates the current position of the root tip. The root growth procedure slides this node along the root's internodes to simulate the root's growth. The Current End node coincides with the Root End node when the root is fully grown.

All node types record a fixed co-ordinate in space, so the root can change direction at any node. This allows a change in direction of the root at a "New Root" node without the need for a "Bend" node. Between each node the root is a straight line. The translation from the physical root system to the list structure is shown in Figure 3.1. The "amount", "level" or "density" of fungal infection used by the model (as represented by the Fungal nodes in Figure 3.1) may be in any appropriate unit, fungal mass / root mass, hyphal length / root surface area, etc. The value is intended to represent the vigour of the fungal lesion, so an ideal unit would be active fungal cytoplasm / root mass.



Root 1		Root 2		Root 3	
BN0	Root Start				
BN1	Branch point	BN1	Root Start		
BN2	Branch point	FN0	Fungal increase	BN2	Root Start
EN0	Root end	FN1	Fungal decrease	PN0	Root bend
		EN1	Root end	EN2	Root end

Figure 3.1: Translation from physical root system to list representation.

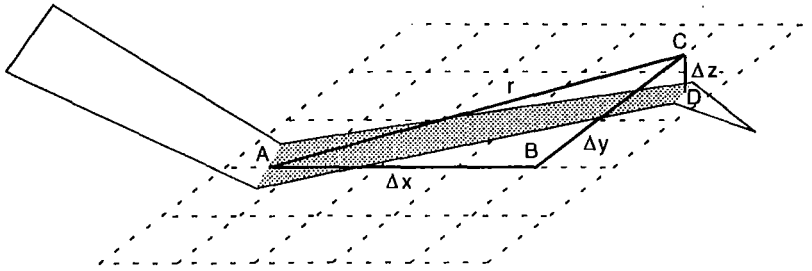
Root position

It is convenient to define the term "root position" to refer to the 1-dimensional coordinate system identifying a node's position on a root. This may be normalised (0..1), or in the range $0..l$, where l is the length of the root. The proximal end of the root is taken as the root position origin.

All nodes possess a root position coordinate, with the node representing the end of the root occurring at l.

Direction

The notation of Henderson et al. [27] is used to describe the direction in which a root is growing. "Azimuth" refers to the horizontal, or compass direction, while "angle" refers to the deviation from the horizontal. (Figure 3.2).



$$
\begin{aligned}
^{*}\text{azimuth} &= \arctan(\Delta y/\Delta x) \\
\text{angle} &= \arctan(\Delta z/r) \\
r &= \sqrt{\Delta x^2 + \Delta y^2}
\end{aligned}
$$

(*see section 5.11 for calculation)

Figure 3.2: Angle / azimuth notation for the shaded internode's direction of travel. The triangle ABC is in the horizontal plane, the line segment CD is vertical and below the plane of ABC. The root intersects the plane at A then descends to D

3.2.2 Records – information packets

Because of the large number of variables required to describe a root system, the concept of "sets" of related variables is useful. The term "record" is adopted for this purpose. The following records are used in the models representation of the root system.

Stem record There is one stem record for each tree. A stem record may include the following values:

x, y, z	The coordinates of the tree within the stand.
type	The type of tree, in the armillaria pathosystem considered here, this will invariably be *Pinus radiata*.
$root_0$	The root record (see below) for this tree's main root, or root collar.
time	The time (relative to simulation time 0) at which the tree was planted.
age	The physiological age of the tree at planting.

Root record There is one root record for each root. The root record contains information that applies to the whole root. A root record may include the following values.

x, y, z	The coordinates of the start (proximal end) of the root.
$length_c$	The current length of the root.
$length_f$	The final length of the root when mature.
phase	Current physiological growth phase of this root (see section 3.5).
$time_c$	Time at which this root was created.
$time_p$	Time at which this root advances to the next physiological growth phase.
parent	The root (record) from which this root originated.
$node_0$	The first node record (see below) for this root.

Node record A node record may include the following values.

x, y, z	The coordinates of the node.
position	The root position of the node on its root (see section 3.2.1).
type	The type of node, this determines what the node represents, it may be the end of the root, the beginning or end of a fungal lesion, a branch point to a higher order root, or simply a node to indicate (via its coordinates) the shape of the root (see section 3.2.1).
root	If the node indicates a branch point, the record will include a reference to the root (record) for the new root.
$fungus$	If the node indicates a fungal lesion, it will also indicate which type of fungus is involved.
$fungus_\Delta$	If the node indicates a fungal lesion, it will also indicate by what amount the fungal infection increases or decreases at this node.
next	The next node on this root, unless this node marks the end of the node.

3.2.3 Root diameter

The representation of root diameter is also possible. A diameter could be specified for the beginning of each root, and a tapering function used to model specific diameter at any point along the root. If greater flexibility were required, a diameter could be specified at each node, and a tapering function used only between one node and the next. In the application of the method to the *Pinus radiata* / armillaria pathosystem, no direct root diameter estimate is required. Root diameter is largely insignificant when compared to inter-tree distances, and is incorporated indirectly, but adequately, by the "radius of infectivity" value used in the modelling of cross-infection events.

3.3 Root architecture - dynamic model processing

3.3.1 Status record

In most of the various analyses performed by the simulation, information in addition to that supplied by the "static" stem, root and node records described in section 3.2.2 is required. As each internode of each root is examined, it is necessary to know (1) the type of tree the internode belongs to, (2) the root's fungal load, (3) the state of the root system nearby, and (4) the internode's direction of travel. All this information could be stored in each node record, but that would be extremely inefficient, as the information is merely a function of other values already stored. Instead, a "status record" is used. The status record is essentially a collection of information about root system at a specific place and time. The status record may include the following values, only some of which will be relevant to any given analysis.

Status record

x, y, z	The coordinates of the place in the root system to which the status record applies.
$order$	the order of the root to which the status record applies.
$angle$	the angle of the internode
$azimuth$	the azimuth of the internode.
$fungi_{mod}$	Modification factor for plant/fungal and fungal/fungal interactions from seasonal and environmental effects.
$fungi_1$	The level of various fungal populations at the point to which the status record applies.
$fungi_2$	The mass (length $\times$ level) of any fungal populations that the point to which the status record applies lies within.
$fungi_3$	The mass of any fungal populations between the point to which the status record applies and the stem.

The last three components are illustrated in Figure 3.3. At the node P, $fungi_1$ is $d1$, $fungi_2$ is $d1(l4 + l5) + d2 \times l3$ and $fungi_3$ is $d1(l1 + l4) + d2(l2 + l3)$. The advantage of the status record is that only one is required, it is part of the analysis, rather than of the root system.

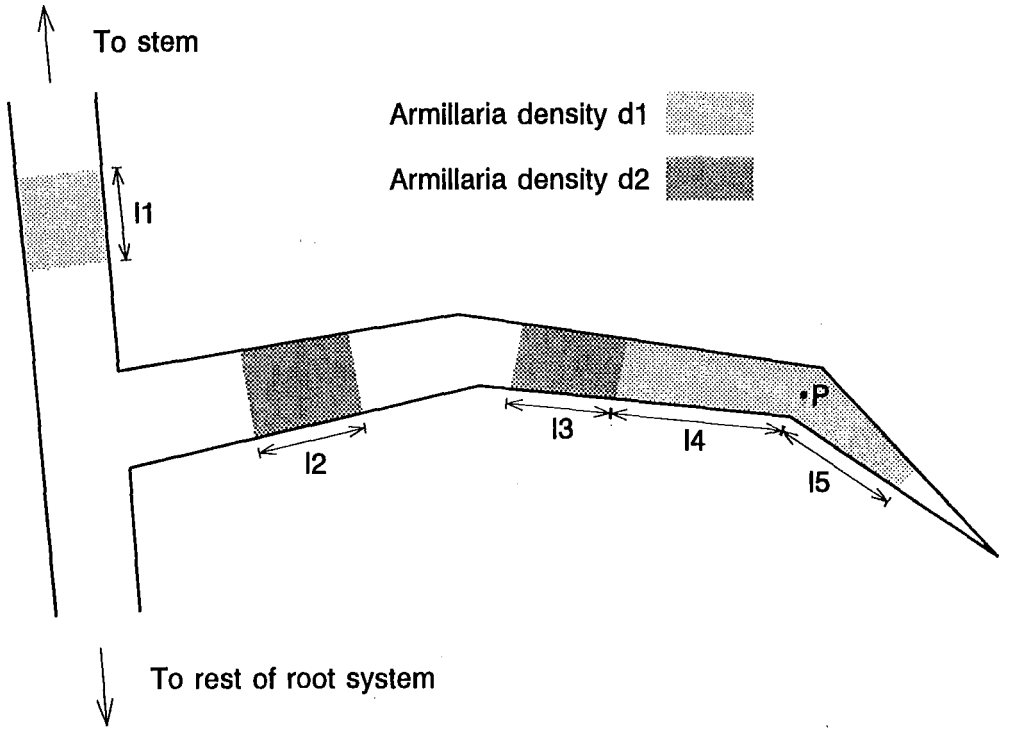


Figure 3.3: Fungal status information stored in a "status record".

3.3.2 Status filter

In many cases it is useful to isolate a subset of roots or internodes (root sections) based on some arbitrary criteria. The simplest case arises when roots of only one order are of interest. More complicated conditions may also be useful, for example it may be necessary to isolate armillaria infected sub-sections of mature second order roots without significant trichoderma activity to estimate the number of possible rhizomorph initiation sites.

Most of these root system states can be described by defining a minimum and maximum status record. Only those internodes whose own status records lie between the minimum and maximum are selected. The following example illustrates a minimum-maximum pair of status records that could be used to isolate live, deep, second or third order roots sections with both armillaria and trichoderma populations.

Record component	Minimum record	Maximum record
x, y, z coordinate	–100,–100,–100	100,100,–1.5
$order$	2	3
$phase$	Growing	Senescent
$angle$	$-\pi$	π
$azimuth$	$-\pi$	π
$fungi_1$–armillaria, level of infection at this point	0.01	100
$fungi_2$–armillaria, "mass" (length × level) of this lesion	10	100
$fungi_1$–trichoderma, level of infection at this point	0.01	100
$fungi_2$–trichoderma, "mass" of this lesion	10	100
$fungi_3$, all fungal species between here and the stem	0	100

Components of the status record that aren't relevant are given "all inclusive" ranges, in this example no point within ± 100 m of the simulation x, y origin will be excluded, but points must be deeper than 1.5 m ($-100 \leq z \leq -1.5$). Similarly the $angle$, $azimuth$, and $fungi_3$ values will not exclude any part of a root system.

3.3.3 Analysis procedure

Having defined the "records", or sets of information required to characterise the root system, any internode (section) of the root system can be described by a stem record, giving general information about the tree, a root record, giving more specific information about the root, a node record for both the proximal and distal ends of the internode, giving highly localised information about this root section, and a status record, giving information derived from the root system between the stem and the root.

For example, an analysis of the root system in Figure 3.4 would require the characterisation of internodes (root sections) I1 and I2. Assume the root system is 2-dimensional and lies in the plane $y = 3$. The descriptions of internodes I1 and I2 would share the same stem record, which,

among other details listed above, will list the stem's x, y, z coordinate as $4, 3, 0$. The internodes could be described by the following root, node, and status records.

	Internode I1	Internode I2
Root record		
x, y, z	2.00, 3.00, −0.06	2.17,3.00,-0.07
$length_f$	1.006	0.748
$length_c$	1.006	0.748
$phase$	Mature	Growing
$time_c$	20	200
$time_p$	100000	2000
$node_0$	(a)	↓
Start node record		
x, y, z	1.81, 3.00, −0.49	2.21,3.00,-0.37
$position$	0.472	0.303
$type$	Bend	Fungal
$fungus$	N/A	Armillaria
$fungus_\Delta$	N/A	+10
$next$	↓	↓
End node record		
x, y, z	1.92, 3.00, −0.72	2.23,3.00,-0.57
$position$	0.727	0.502
$type$	Branch	Bend
$root$	(d)	N/A
$next$	(b)	(c)
Status record		
$order$	1	2
x, y, z	2.00, 3.00, −0.06	2.17,3.00,-0.07
$angle$	−1.16	−1.45
$azimuth$	0	0
$fungi_1$	0	10
$fungi_2$	0	3.31†
$fungi_3$	2.12‡	0

†10× length of internodes I2 and its successor 6 & 8
‡from internode starting at (a)

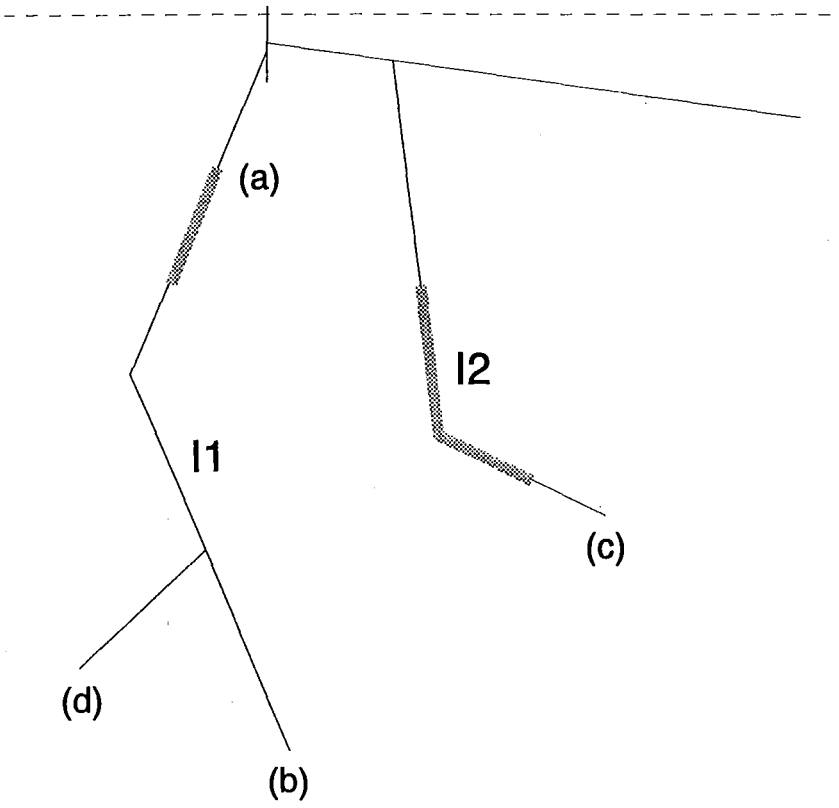


Figure 3.4: Root system analysis.

To determine, for example, the proportion of the root system infected by armillaria, the information (stem, root, start-node, end-node and status records) for all internodes would be considered, and the sum of the lengths (distance between the x, y, z components of the start- and end-nodes) of all infected internodes ($fungi_1$ in the status record $>$ 0) would be divided by the sum of the lengths of all internodes. Similarly, fungal lesion expansion could be simulated by considering all internodes, and, in the case of fungal nodes, moving the node in the appropriate direction, by adjusting the node record's x, y, z and $position$ components.

3.3.4 Parser

A "parser" entity is included to simplify the examination and alteration of the set of records that represent the root system. The parser processes the records and presents the "client" (for example, an analysis routine determining the proportion of the root system infected) with the stem, root, start-node, end-node and status records for each internode.

An example of the parsers action for the root system is given in Figure 3.5:

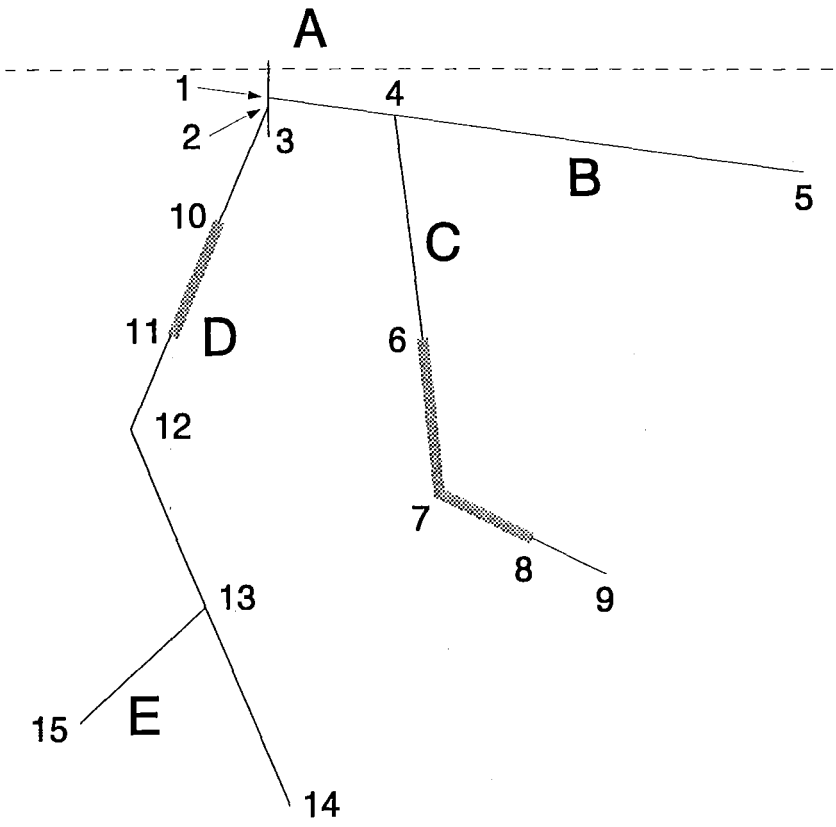


Figure 3.5: Root system parser, example data.

A possible Stem record, which is shared by all internodes, is as follows:

x, y, z	2,3,0
type	*Pinus radiata*
$root_0$	Root A
time	0 (tree planted at start of simulation)
age	0 (seedling, physiological age = "ground" age)

The parser would supply the client routine with fifteen sets of root, start-node, end-node and status records, one for each internode.

For the first internode of each root, the parser could automatically generate a temporary "Root Start" node, rather than have the simulation duplicate the information stored in the "Branch" node on the parent root and in the root record. The angle and azimuth values are re-calculated

between each internode. If the parser is working recursively, an end-node of type "Branch" will cause it to start processing the child root. The *order* component of the status record is incremented appropriately. The fifteen internode descriptions the parser would present to a client process analysing the root system are as follows:

The format used here is

First line	**Root record**
t_c/t_p	$time_c/time_p$, creation time / time of advance to next phase
phase	physiological growth phase
l_c/l_f	$length_c/length_f$, current length / final length
Second line	**Start-node record**
x, y, z	coordinate
p	*position*, root position
type	Node type, and additional, type specific, information
Third line	**End-node record**
x, y, z	coordinate
p	*position*, root position
type	Node type, and additional, type specific, information
Fourth line	**Status record**
order	root order
azi/ang	*azimuth/angle*, direction of internode growth
$f_1/f_2/f_3/f_m$	$fungi_1/fungi_2/fungi_3/fungi_{mod}$, the fungal levels and modification factor (section 3.3.1).

The fifteen internode descriptions, in the order the parser would supply them to the client analysis, are

1 Root A t_c/t_p 0.0/100000.0, Mature, l_c/l_f 0.100/0.100
Node - at 2.00, 3.00, 0.00, p 0.000, Root Start
Node 1 at 2.00, 3.00,-0.05, p 0.050, Branch (Root B)
order 0, *azi/ang* 0.00/-1.57, $f_1/f_2/f_3/f_m$ 0.00/0.00/0.00/1.00

The *order* component of the status record is incremented as the parser moves from root A node 1 to root B.

2 Root B l_c/l_p 10.0/100000.0, Mature, l_c/l_f 0.719/0.719
Node - at 2.00, 3.00,-0.05, p 0.000, Root Start
Node 4 at 2.17, 3.00,-0.07, p 0.171, Branch (Root C)
order 1, *azi/ang* 0.00/-0.14, $f_1/f_2/f_3/f_m$ 0.00/0.00/0.00/1.00

3 Root C t_c/t_p 200.0/2000.0, Growing, l_c/l_f 0.748/0.760
Node - at 2.17, 3.00,-0.07, p 0.000, Root Start
Node 6 at 2.21, 3.00,-0.37, p 0.303, Fungal Armillaria +10
order 2, *azi/ang* 0.00/-1.45, $f_1/f_2/f_3/f_m$ 0.00/0.00/0.00/1.00

4 Root C t_c/t_p 200.0/2000.0, Growing, l_c/l_f 0.748/0.760
Node 6 at 2.21, 3.00,-0.37, p 0.303, Fungal Armillaria +10
Node 7 at 2.23, 3.00,-0.57, p 0.502, Root Bend
order 2, *azi/ang* 0.00/-1.45, $f_1/f_2/f_3/f_m$ 10.0/3.31/0.00/1.00

The parser adjusts $fungi_1$ and $fungi_2$ to the level and size of the lesion between nodes 6 and 8.

```
5  Root C tc/tp 200.0/2000.0, Growing, lc/lf 0.748/0.760
   Node 7 at 2.23, 3.00,-0.57, p 0.502, Root Bend
   Node 8 at 2.35, 3.00,-0.62, p 0.634, Fungal Armillaria -10
   order 2, azi/ang 0.00/-1.45, f1/f2/f3/fm 10.0/3.31/0.00/1.00

6  Root C tc/tp 200.0/2000.0, Growing, lc/lf 0.748/0.760
   Node 8 at 2.35, 3.00,-0.62, p 0.634, Fungal Armillaria -10
   Node 9 at 2.45, 3.00,-0.67, p 0.748, Root End
   order 2, azi/ang 0.00/-0.44, f1/f2/f3/fm 0.00/0.00/3.31/1.00
```

$fungi_3$ now identifies the internode as being separated from the stem by the fungal mass between nodes 6 and 8. As the end-node is a "Root End" node, the parser resumes the suspended processing of root B.

```
7  Root B tc/tp 10.0/100000.0, Mature, lc/lf 0.719/0.719
   Node 4 at 2.17, 3.00,-0.07, p 0.171, Branch (Root C)
   Node 5 at 2.71, 3.00,-0.15, p 0.719, Root End
   order 1, azi/ang 0.00/-0.14, f1/f2/f3/fm 0.00/0.00/0.00/1.00
```

Again, the end-node is a "Root End", so the parser resumes processing root A.

```
8  Root A tc/tp 0.0/100000.0, Mature, lc/lf 0.100/0.100
   Node 1 at 2.00, 3.00,-0.05, p 0.046, Branch (Root B)
   Node 2 at 2.00, 3.00,-0.06, p 0.060, Branch (Root D)
   order 0, azi/ang 0.00/-1.57, f1/f2/f3/fm 0.00/0.00/0.00/1.00
```

Here root D begins.

```
9  Root D tc/tp 20.0/100000.0, Mature, lc/lf 1.006/1.006
   Node - at 2.00, 3.00,-0.06, p 0.000, Root Start
   Node 10 at 1.93, 3.00,-0.22, p 0.175, Fungal Armillaria +10
   order 1, azi/ang 3.14/-1.16, f1/f2/f3/fm 0.00/0.00/0.00/1.00

10  Root D tc/tp 20.0/100000.0, Mature, lc/lf 1.006/1.006
    Node 10 at 1.93, 3.00,-0.22, p 0.175, Fungal Armillaria +10
    Node 11 at 1.86, 3.00,-0.42 p 0.387, Fungal Armillaria -10
    order 1, azi/ang 3.14/-1.16, f1/f2/f3/fm 10.0/2.12/0.00/1.00
```

All internodes below 10–11 in the root hierarchy will have a $fungi_3$ value of 2.12, reflecting their separation from the stem by this fungal mass.

```
11  Root D tc/tp 20.0/100000.0, Mature, lc/lf 1.006/1.006
    Node 11 at 1.86, 3.00,-0.42, p 0.387, Fungal Armillaria -10
    Node 12 at 1.81, 3.00,-0.49, p 0.472, Root Bend
    order 1, azi/ang 3.14/-1.16, f1/f2/f3/fm 0.00/0.00/2.12/1.00
```

```
12  Root D tc/tp 20.0/100000.0, Mature, lc/lf 1.006/1.006
    Node 12 at 1.81, 3.00,-0.49, p 0.472, Root Bend
    Node 13 at 1.92, 3.00,-0.72, p 0.727, Branch (Root E)
    order 1, azi/ang 0.00/-1.15, f1/f2/f3/fm 0.00/0.00/2.12/1.00

13  Root E tc/tp 1500.0/3500.0, Growing, lc/lf 0.227/0.320
    Node - at 1.92, 3.00,-0.72, p 0.000, Root Start
    Node 15 at 1.75, 3.00,-0.87, p 0.227, Root End
    order 2, azi/ang 3.14/-0.74, f1/f2/f3/fm 0.00/0.00/2.12/1.00

14  Root D tc/tp 20.0/100000.0, Mature, lc/lf 1.006/1.006
    Node 13 at 1.92, 3.00,-0.72, p 0.727, Branch (Root E)
    Node 14 at 2.03, 3.00,-0.98, p 1.006, Root End
    order 1, azi/ang 0.00/-1.15, f1/f2/f3/fm 0.00/0.00/2.12/1.00

15  Root A lc/lp 0.0/100000.0, Mature, lc/lf 0.100/0.100
    Node 2 at 2.00, 3.00,-0.06, p 0.061, Branch (Root D)
    Node 3 at 2.00, 3.00,-0.10, p 0.100, Root End
    order 0, azi/ang 0.00/-1.57, f1/f2/f3/fm 0.00/0.00/0.00/1.00
```

The "recursive" behaviour (as the parser invokes itself to process new roots at branch points) in this example is only one possible parser algorithm. Alternatively, the parser could make lists of the Branch type nodes it encounters, and only process them after it has completely processed the root it is currently parsing.

3.4 Simulated root system construction

3.4.1 Parameter matrix

Root oriented spatial models like that of Henderson et al. [27] require a large number of variables to model tree root architecture. Different probability density functions, such as exponential, gamma, poisson, uniform, may be used to model the same variable for different orders of root. To model occurrence, position, azimuth and angle for both branching and bending as well as forking and length for six orders of roots, Henderson et al. [27] could have used 60 $((4\times2+2)\times6)$ completely separate statistical distributions. In practice the same distributions were used for some variables (for example roots of all orders were considered to bend $Poisson(0.055)$ times), and other distributions varied in only a single parameter, that being the order of the root to which they were applied.

In the model developed here, the number of variables is even greater. Root life span, root

growth rate, occurrence, position and size of fungal lesions, and elongation rate of fungal lesions must be added to the architectural variables modelled by Henderson et al. [27]. When different physiological stages and different species of fungi are considered, the total number of statistical distributions (variables) for a $n = 6$ order root system with $p = 6$ physiological stages and $f = 3$ types of fungi, is $n(4 + 9p + 6pf)$, or 996 (Table 3.1).

Characteristic	Variables	Levels
Length	length[b]	$n \times p^{a}$
Lifespan	duration	$n \times p$
Growth rate	rate	$n \times p$
Fungal elongation	rate	$n \times p^{a} \times f$
Branching	number[b], position, 2×(angle, azimuth)[d]	$n \times p^{a}$
Bending	number, position, angle, azimuth	n
Fungal lesions	number, position, angle[c], azimuth[c], size	$n \times p^{a} \times f$

n - number of root orders
f - number of species of fungi
p - number of physiological stages

[a]Not applied here
[b]May depend on parent's physiological stage
[c]Not in *Pinus radiata* system
[d]Both parent and child angle and azimuth

Table 3.1: Root system variables.

For simplicity of both notation and implementation (section 5.3), the statistical functions may be grouped as:

$$F_{t,o,f,a,p,s}$$

where

t = type of tree
o = order of root
f = feature (Table 3.2)
a = attribute (Table 3.2)
p = physiological stage
s = species of fungi

For example, $F_{Pine,2,Branch,Position,Mature,--}$ may refer to a uniform variate in the range 0 to 1, while $F_{Pine,5,Fungi,Number,Growing,Armillaria}$ may be invariably zero, if armillaria is found

not to colonise such high order roots. The relationships between "features" and "attributes" are shown in Table 3.2. The $BVect$ feature is the initial direction of a child root relative to its parent root. $BVect$ could be collapsed into the Root feature, but is listed separately to avoid ambiguity with the change in direction of the parent root at the point of branching.

	Attributes						
	Length	Elongation	Number	Position	Angle	Azimuth	Lifespan
Features							
Root	o		o				o
Branch			o	o	o	o	
BVect					o	o	
Bend			o	o	o	o	
Fungi	o	o	o	o	o	o	o

Table 3.2: Root variable matrix form.

3.4.2 Constructor

The type of tree, and its position, will be defined by the user. For example, a root system may be constructed from the initial information that a tree of type $Pine$ has its root collar, or 0th order root, at x_0, y_0, z_0. The direction of the 0th order root at its proximal end will usually be straight down ($angle = -\pi/2$, $azimuth$ undefined), but any direction is valid.

To add a root of order O, given that the x, y, z coordinate and direction of growth ($angle$ and $azimuth$) of it proximal end are known the following procedure is followed:

1. Place a "Root End" node on the root, determining the length of the root, and thus the "Root End" node's root position, from the statistical distribution $F_{Pine,O,Root,Length,--,--}$. The node's x, y, z position will be undefined at this stage.
2. Determine the number of "Branch" nodes on the root from $F_{Pine,O,Branch,Number,--,--}$. Place this many "Branch" nodes on the root, determining their root space position from $F_{Pine,O,Branch,Position,--,--}$. Their x, y, z positions will be undefined at this stage.
3. Determine the number of "Bend" nodes on the root from $F_{Pine,O,Bend,Number,--,--}$. Place this many "Bend" nodes on the root, determining their root position from

$F_{Pine,O,Bend,Position,--,--}$. Their x, y, z positions will be undefined at this stage.

4. For each species (sp) of fungi to be considered, determine the number of lesions to appear on the root from $F_{Pine,O,Fungi,Number,--,sp}$. For each lesion add two nodes to the root, one at $F_{Pine,O,Fungi,Position,--,sp}$ marking an increase in the level of sp, and one $F_{Pine,O,Fungi,Length,--,sp}$ further "down" the root, marking an equivalent decrease in sp. If the root position of this second node places it beyond the end of the root, adjust it to fit, or re-sample all the distributions until the lesion fits within the root.

5. Working from the proximal end of the root, calculate the x, y, z position of each node from the root position of the node and the $angle$ and $azimuth$, altering $angle$ and $azimuth$ with $F_{Pine,O,node-type,Angle,--,(sp)}$ and $F_{Pine,O,node-type,Azimuth,--,(sp)}$ between each calculation.

The process is repeated for the $O+1$ order roots at each "Branch" node, their initial direction of travel being defined by $F_{Pine,O,BVect,Angle,--,sp}$ and $F_{Pine,O,BVect,Azimuth,--,sp}$. Depending on the fungus to be modelled, it may be necessary to add a "Fungal" node to any child roots branching from infected sections, to completely delimit fungal lesions (Figure 3.6).

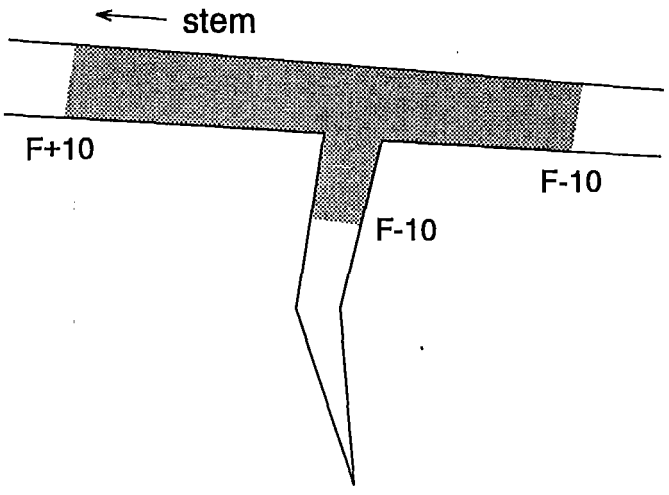


Figure 3.6: Nodes required to delimit fungal lesion, assuming a fungal density score of 10.

Phase	Description
Future	A root that doesn't exist yet, i.e. one occurring past the Current End node of its parent.
Waiting	The Current End node of the parent has passed this point, but the root is not yet growing. Allows additional first order roots to appear on the root bole several years after the tree starts to grow.
Growing	Actively elongating, child roots that die will be replaced
Mature	Currently equivalent to Growing.
Senescent	Retains some resistance to fungal attack, but is not elongating, and has no active child roots (i.e. all child roots will also be Senescent or older).
Dead	Reduced resistance to fungal attack, negative elongation may represent decay of distal end. Prime site for rhizomorph initiation.
Decayed	In an advanced state of decay, won't support rhizomorphs, but may still be infective given direct contact.

Table 3.3: Root physiological phases.

3.5 Representation of root physiology

As shown in Table 3.3, a variety of root physiological stages are considered. The combination of a root's physiological stage, it's position within the soil, and it's order largely define it's behaviour. The typical duration of each of these phases is a required input of the model, local conditions may increase or decrease the time spent in each phase.

3.6 Fungal behaviour

3.6.1 Nodes

As previously indicated (Figure 3.1), the infection of a section, or internode, of a root can be represented by a pair of nodes. The first (in proximal $\rightarrow$ distal order) node denotes an increase in the level of infection, usually from zero, the second node denotes a decrease, usually to zero. If lesions are allowed to overlap, nodes may increase / decrease infection levels from non-zero values. An attenuating lesion could be represented by a proximal node incrementing the level of infection by, for example, 20, followed by a pair of separate nodes each decreasing the infection

level by 10 (Figure 3.7). This approach allows lesions of different densities, and of different fungal species, to overlap.

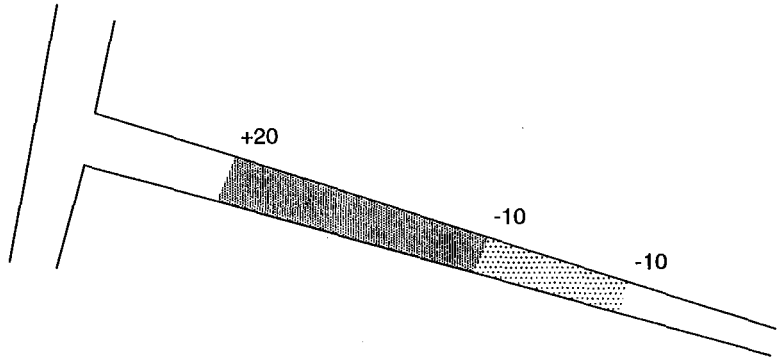


Figure 3.7: An attenuating fungal lesion.

3.6.2 Elongation

The elongation (or contraction), of fungal lesions can be represented by "sliding" the fungal nodes up and down the root. A lesion expands when its proximal node moves "up" the root towards the stem, and/or its distal end moves "down" the root towards the root tip. As these mobile nodes encounter other nodes, further modifications to the root system may be required. Figure 3.8 illustrates the possible node interactions:

- Fungal node 3 grows up past bend node 2. The nodes swap their positions on the list of nodes for the root (3 now occurs before 2), but no other change occurs.
- Fungal node 3 reaches branch node 1. Assuming the fungus spreads from the child root to the parent, node 3 ceases to exist, and two new nodes are created on the parent root, either side of node 1.
- Fungal node 4 passes branch node 5. A node must be added to the child root to mark the limit of the lesion (Figure 3.6).
- Fungal node 4 passes fungal node 6. Depending on the behaviour of the fungus being modelled, three responses are possible.
  - If the fungal densities indicated by nodes 3 and 6 are the same, consider the lesions to have merged. Nodes 4 and 6 cease to exist.

- If the densities differ, remove nodes 4 and 6, and replace them with a node marking the change in density from the upper to the lower lesion.
- If multiple lesions can exist in the same section of root, allow the lesions to overlap. The fungal density in the internode between nodes 6 and 4 will be the sum of the densities occurring in the individual lesions.

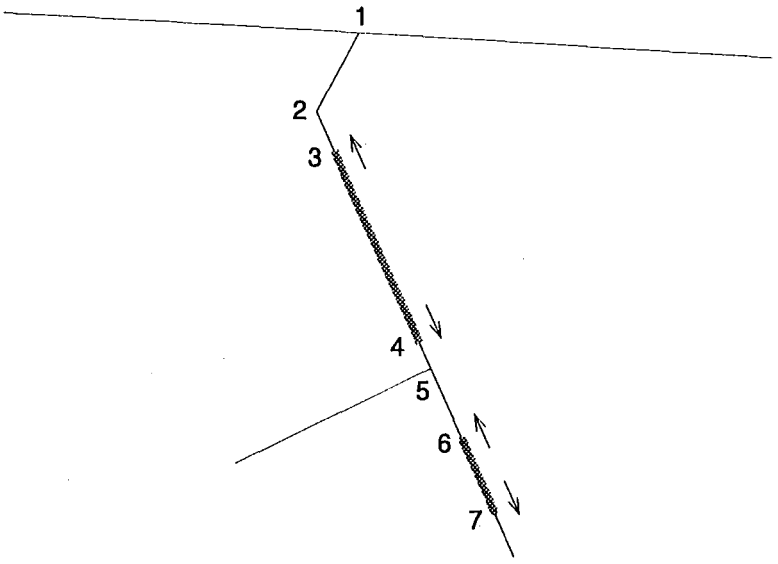


Figure 3.8: Fungal lesion expansion.

3.6.3 Free soil fungi and roots

Some soil fungi are less closely associated with plant roots than others. To represent loosely associated fungi, such as trichoderma, spherical regions can be defined as being occupied by the fungus. Determining what part, if any, of an internode lies within a sphere is not difficult. Rather than adding a list of spherical regions and a need to constantly check for intersection to the simulation, the spherical regions can be used to add "lesion" delimiting node pairs to roots, as is done for more closely associated fungi (section 3.6.1). Figure 3.9 illustrates how spheres can define points (nodes) bracketing the area influenced by free soil fungi.

3.7 Rhizomorphs

These branching exploratory hyphal growths are structurally very similar to a tree root system. They can be represented in the same way, with another set of statistical distributions of the form $F_{Rhiz.,O,Branch,Number,--,--}$, $F_{Rhiz.,O,Root,Length,--,--}$, etc. As previously discussed (section 3.3.2), a "status filter" can be used to isolate root sections suitable for producing rhizomorphs. The root section must be large enough to act as a food source for the rhizomorph. Often the root from which a rhizomorph originates will be in an advancing state of decay, and the rhizomorph may die from a lack of nutrients as a result. To allow the simulation to remove rhizomorphs whose root has decayed, a special type of fungal node is used to mark the place on the root system where the rhizomorph originates. Whereas other fungal nodes record a species and a density increment (or decrement), the rhizomorph initiation fungal node records a reference to the "stem" record of the rhizomorph it represents.

3.8 Fungal interaction

Depending on the fungi involved, the following interactions are possible:

1. **Spatial exclusion.** Ectomycorrhizal fungi form dense sheaths on root surfaces, and so may prevent other root/fungi interactions. The simulation's use of fungal nodes to represent root sections infected with armillaria can also represent this phenomenon, without significant modification.

2. **Resource competition.** Many soil fungi compete for the same resources; carbon sources in particular are a limiting factor. A strong saprophyte population may limit disease virulence by reducing the availability of residual carbon in the soil, limiting the rate of spread if disease organisms from one host to another, and reducing the persistence of disease organisms in the absence of the host. Although not part of the current implementation, a spatially oriented simulation could model this effect by maintaining a three-dimensional grid representing concentrations of different resources.

3. **Parasitism.** Some soil fungi actively parasitise other fungal species, with exploratory hyphae growing into close contact with the hyphae of the host species, penetrating the

host's cell walls with chitinase and similar enzymes, and consuming host cytoplasm. The distribution of the host species will often be influenced by the distribution of plant roots, so the availability of a detailed root map would allow such interactions to be modelled with more detail than is normally possible. Depending on their behaviour, host species could either be modelled as tree-like branching structures as rhizomorphs are (section 3.7), or as point sources.

4. **Antibiotic production.** Many soil microorganisms produce compounds that inhibit or kill other species. Antibiotics may be produced in response to direct aggression from other species (possibly also in the form of antibiotics), in response to resource limitation (indicating overcrowding), or, as appears to be the case with trichoderma, in an apparently non-specific attempt to reduce competition from other species [29]. Antibiotics may spread through the soil by diffusion processes, and existing rhizosphere diffusion models may be incorporated into a spatial simulation framework to model the volume affected [18].

5. **Synergism.** A combined attack on plant defences by more than one fungal species may succeed where either species alone may fail. Also, some soil microorganisms are dependent on ambient supplies of vitamins and other complex nutrients that may leak into the soil solution from organisms capable of synthesising them from simpler molecules. This may lead to pairings of organisms. As in other cases, the critical factor is location, and a spatial simulation is well equipped to represent such interdependence.

3.9 Infection

3.9.1 Infection score

Close physical proximity is an obvious prerequisite for the spread of a fungus (disease or symbiont) from one root (or rhizomorph) to another. But whether or not a root coming into contact with diseased material will become infected is a function of the root's physiological state, the environmental conditions (including the presence of other organisms), and the vigour of the disease material. An "infection score" is used to simulate this interaction. Infection occurs if the root's score (S_r), is lower than armillaria's score (S_a). Scores are calculated as follows:

$$S_a = L_{S(a)} \times M_{S(a)} \times M_{D(a)} \tag{3.1}$$

where

$L_{S(a)}$ = level of armillaria at source

$M_{S(a)}$ = modifier for armillaria at source

$M_{D(a)}$ = modifier for armillaria at destination

$$S_r = 1 \times L_{S(t)} \times M_{S(t)} \text{ if } L_{S(t)} \neq 0 \times L_{D(t)} \times M_{D(t)} \text{ if } L_{D(t)} \neq 0 \tag{3.2}$$

where

$L_{S(t)}$ = level of trichoderma at source

$L_{D(t)}$ = level of trichoderma at destination

$M_{S(t)}$ = modifier for trichoderma at source

$M_{D(t)}$ = modifier for trichoderma at destination

The modifiers used in the calculation of the scores reflect the innate characteristics of the roots on which the interaction is occurring. The resistance to infection by armillaria will obviously vary between Mature first order roots, Mature fourth order roots, and Senescent first order roots. Before application, the modifiers themselves may be adjusted by physical environment variables. For example, low soil moisture levels may increase the armillaria modifiers, to reflect increased infection rates under drought stress.

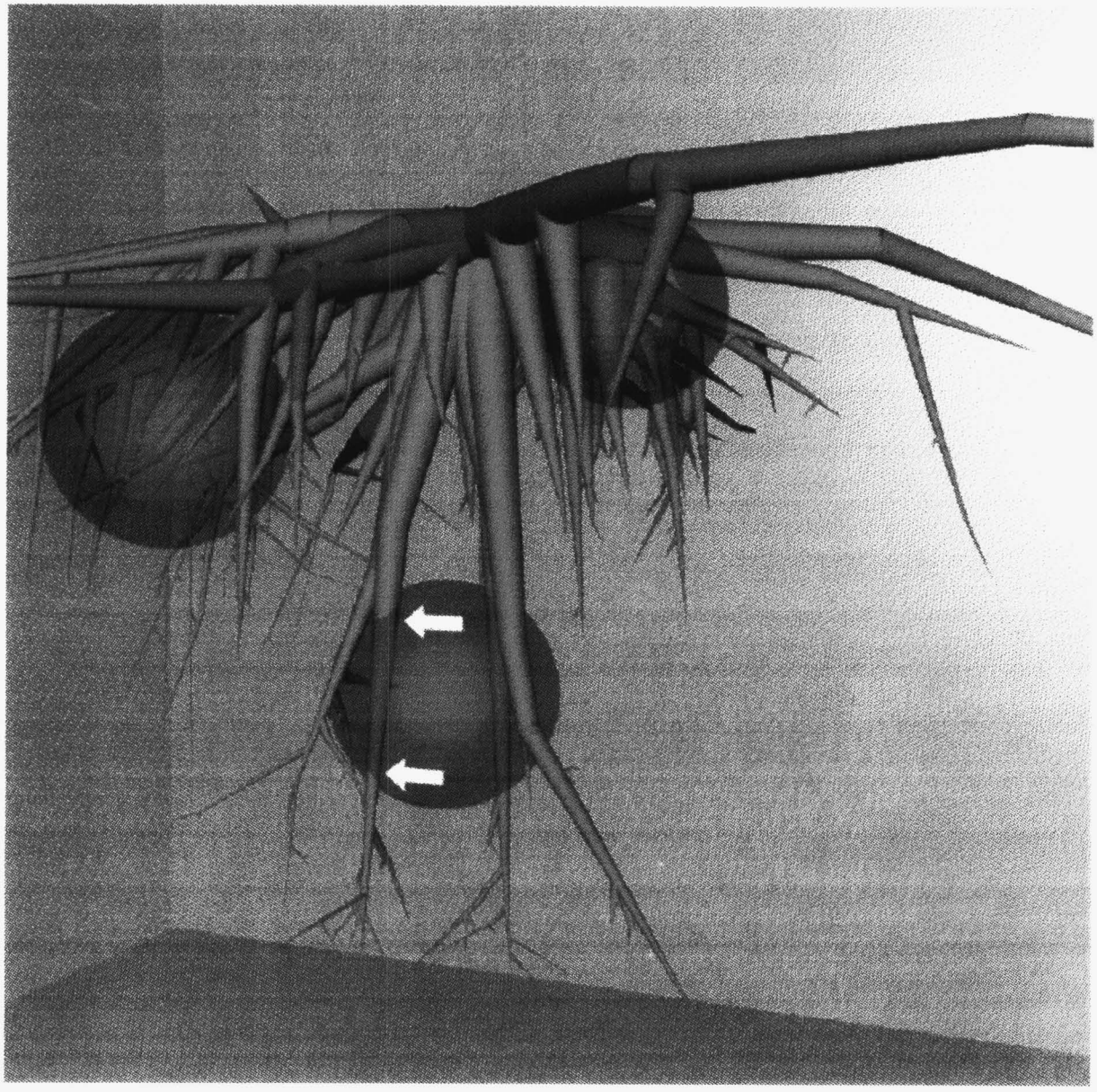


Figure 3.9: A raytrace illustrating the delimiting of internodes influenced by free soil fungi. Fungi increment and decrement nodes can be added at the points indicated to mark the area effected.

Chapter 4

Parameter evaluation

4.1 Derivation of distributions and parameters for system geometry

The many probability distributions required by the model must be obtained from field work that has specifically targeted root system morphology. Most root system studies are biomass oriented and provide data only marginally relevant to spatial questions. One notable exception is Watson and O'Loughlin [62], who excavated and mapped intact root systems of *Pinus radiata* by sluicing. This valuable data set consists of plan and front elevation drawings for 5 trees in each of 3 age groups, 8, 16 and 25 years.

Reconstruction of a three dimensional structure from these plan and elevation drawings is difficult. The drawings are quite detailed and thorough, but a slight 'fuzziness' in the correspondence between the plan and elevation representation of each root means locating roots in both views is not always possible. In addition, there are large sections of the root system that cannot be expected to appear in both views, as they will be concealed by roots closer to the point from which the drawings were made.

In an initial attempt to derive the required parameters, the drawings were digitised as collections of points making up "strings" of short line segments, and these digitised structures are very accurate copies of the original drawings, essentially giving the illustrations a machine readable form. Successful trials were conducted with the translation of these data files into the form used by the model. However, the problems outlined above, combined with the lack of correspondence of the line segment "strings" to individual roots, rendered the recapturing of the three dimensional

structure of the root systems directly from these drawings excessively time consuming given the partially incomplete nature of the data it would have yielded. Instead, an indirect method of quantifying the root morphologies represented by these drawings was devised.

The drawings were divided into segments and bands (Figure 4.1), and the total number of roots, root ends and branching points in each section of the cylindrical grid were counted. Initially this was done by laying an appropriately marked sheet of clear perspex over the drawings. For elevation views only half the grid was used. The root systems were later re-digitised in a more structured manner, and these counts were repeated by the simulation software. These figures did not prove directly useful, and the required parameters (section 3.4.1) were obtained by an iterative adjustment process based on visual comparison of the drawings and model output. The digitised root systems did however allow the objective comparison of the simulated and observed roots (section 6.1).

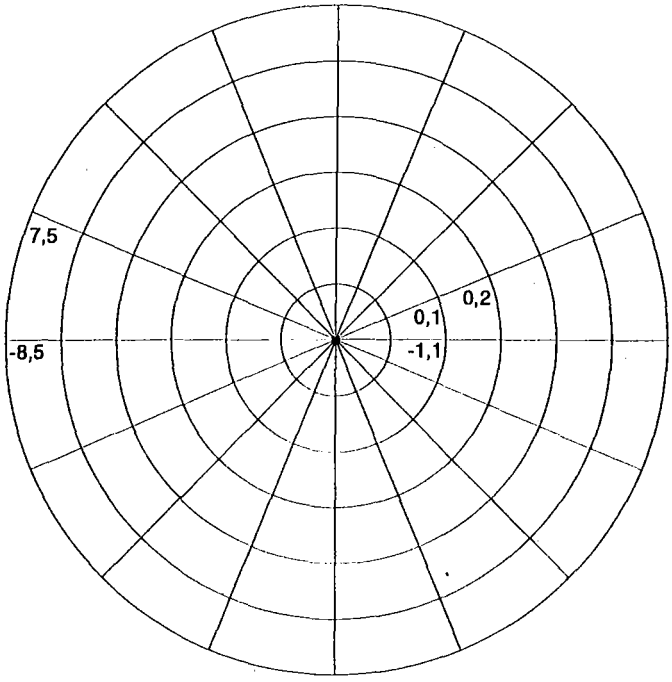


Figure 4.1: The 16×6 circular grid used to form a quantitative description of root system drawings. Numbers indicate the –8 to +7 by 0 to 5 coordinate system used.

4.2 Data acquisition

4.2.1 *Armillaria spp.* distribution in dead root systems

A realistic assessment of the distribution of *Armillaria spp.* in dead root systems of mature *Pinus radiata* is an essential component of the model. The distribution and sources of infection of Aspen stands by *Armillaria mellea* were examined by Stanosz and Patton [54]. They found rhizomorph penetration to be the most frequent form of infection. Of great significance to this study was their observation of a rapid decrease in lesion formation with increasing distance from the stem, suggesting lower *A. mellea* densities may occur in the peripheral root system than in the main structural root system. However their screening technique may not have included finer elements of the root network.

Hood and Sandberg [31] examined rhizomorph distribution in indigenous New Zealand forest soils, and showed statistical clustering of rhizomorphs in some plots, perhaps suggesting growth radiating from a localised area, such as a dead root system. Sampling was for rhizomorph occurrence rather than length, orientation and other more precise geometric distribution parameters. Earlier work [42, 48, 60] found dead root systems of indigenous species to be unequivocally the major source of *Armillaria spp.* in first rotation pine stands, so the root systems of the previous crop will almost certainly be the major source in second rotation stands. However, aerial spores may play a significant role in colonising thinning stumps.

Patton and Riker [44] found no significant relationship between root segment size and rhizomorph production in experiments with sterilised root blocks, with segments as small as 0.25x2 inches producing rhizomorphs. However, larger segments appeared to produce rhizomorphs more rapidly.

Another significant factor in the distribution of dead root systems, and hence the distribution of *Armillaria spp.* within them, is the rate of decay of the root system. It can be assumed that the main stump will remain for several years, but the persistence of root elements less than 50 mm in diameter is less clear, and so the importance of these components as widely distributed sources of *Armillaria spp.* will require careful assessment. As previously noted, O'Loughlin and Watson [43] examined the rate of decrease of tensile strength in roots less than 30 mm in

diameter and noted only rotted wood remained in this class at 14 months, and only bark sheaths at 29 months.

4.2.2 Distribution of sites susceptible to armillaria in *Pinus radiata* seedlings

Just as the points of origin of armillaria in the newly established plantation are important, so are the points at which it may infect seedlings. Stanosz and Patton [54] found frequent rhizomorph penetration of thick, barked roots, with localised decay at the point of infection. Although this work was with Aspen, if similar behaviour is exhibited with pinus, armillaria contact with fine roots may not be as important as contact with structural root system components.

4.3 Two year old *Pinus radiata* root systems: field observations

Between June 1992 January 1994, 5 two to three year old *Pinus radiata* root systems were excavated to allow characterisation of their geometry. The trees were in a Templeton silt loam, which was loose and friable down to a hard sand layer at 0.30 m. The trees were planted in a North-South rip 0.45 m deep, forming a 0.15 m gutter in the sand layer.

4.3.1 Layered excavation

As little was known initially about the size or complexity of the system to be excavated, it was decided to remove the soil from around the roots in 100 mm layers, photographing and then removing the roots between each layer. A heavy wooden frame (Figure 4.2), 3 m square, was used to provide a fixed ground level, and the coordinates for each photograph were measured from graduations along the sides of the frame. This allowed latter reconstruction of the root system by digitising the photographs (Figure 4.3) using a graphics tablet and either an extension of the root system construction algorithms already in use. Computer algorithms could then be written to analyse the digitised data-structure for such properties as mean branch angle, mean inter-branch distance, frequency of branching for each order of root, and other frequency and position related distributions.

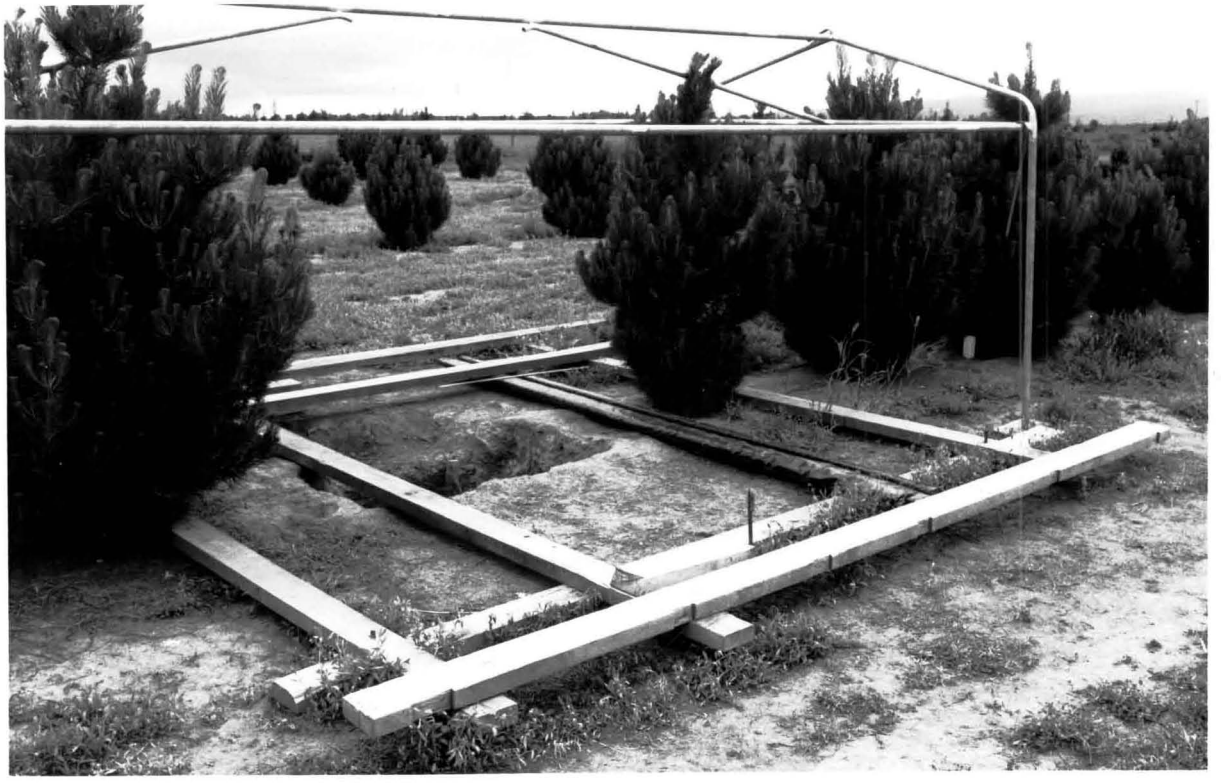


Figure 4.2: Framework for excavation.

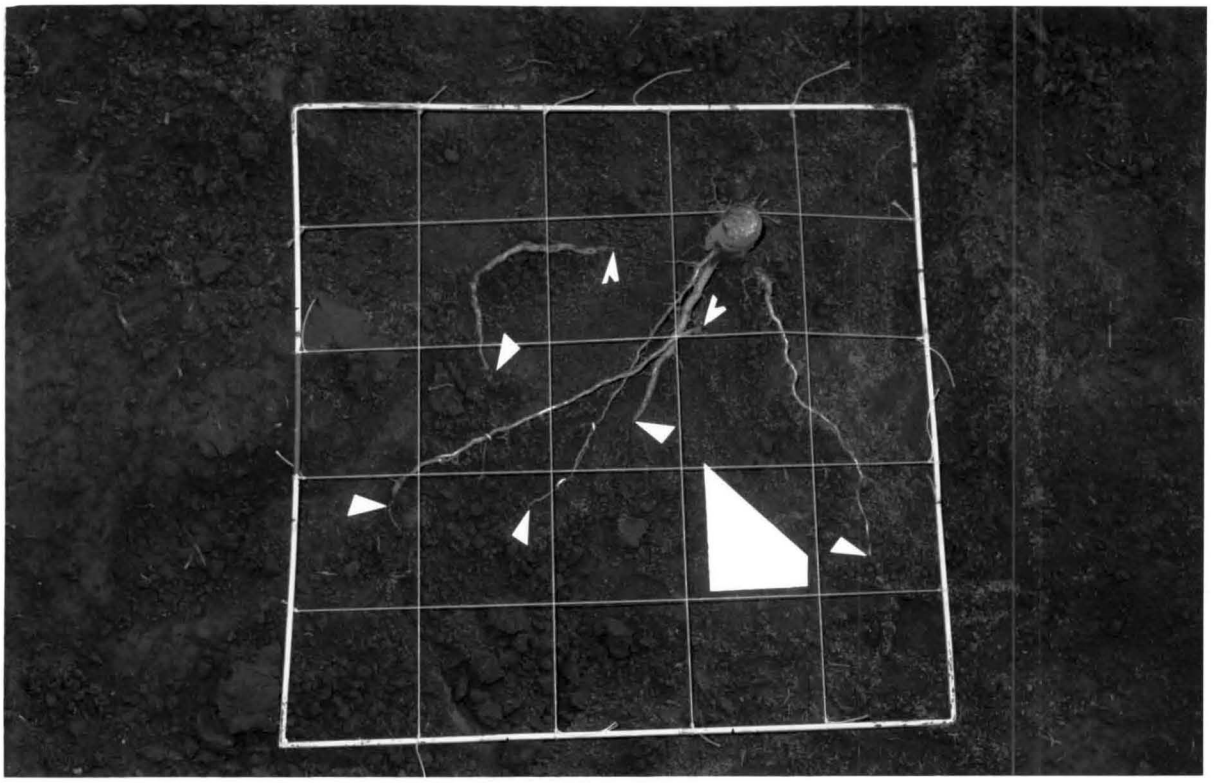


Figure 4.3: Roots in a 100 mm layer.

4.3.2 Stereo Image Pairs

As the root systems were found to be simpler than originally expected, a second method was devised using digitised stereo image pairs. A 180° section of the root system was excavated to a depth of 400 mm, so that only the first-order vertical roots directly below the stump remained in the soil. A pair of images with a separation of 150 mm were recorded using a SVHS video camera supported 1.5 m above the exposed root system (Figure 4.4). Fluorescent green spray paint was applied to the roots to compensate for a low contrast between the roots and the soil background.

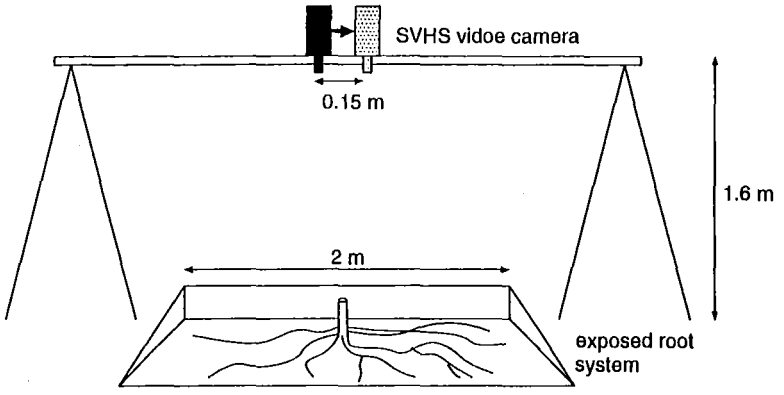


Figure 4.4: Three dimensional image capturing.

The images were transferred to computer readable form using Joyce-Lobel MagiScan equipment, producing a 512 by 512 pixel image in 16 grey levels. The image analysis software supplied with the scanner was used to filter out the dark grey background and leave only the intense white root image. A proprietary computer image manipulation package was then used to colour and merge the left and right images for three-dimensional viewing (Figure 4.5). With or without coloured eye-glasses, the lateral separation between the two images readily illustrated the vertical position of any point in the root system relative to an arbitrarily selected zero level. This information allows the true lengths and angles of the structure to be recorded, unlike a purely two-dimensional view.

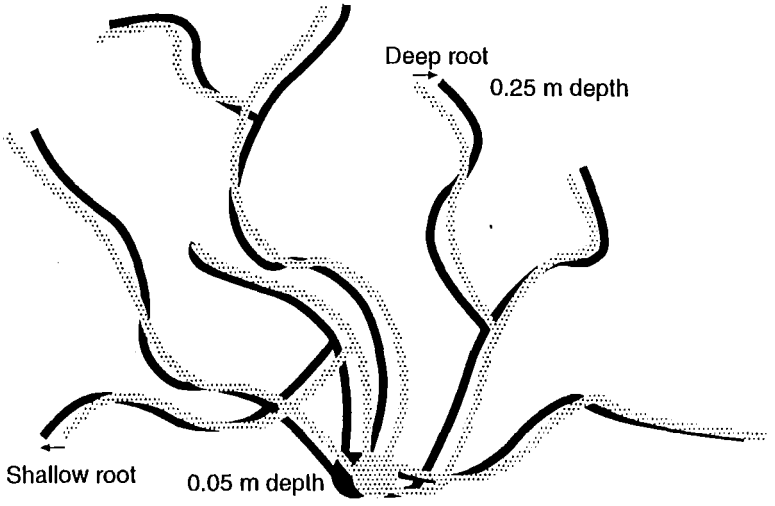


Figure 4.5: Three dimensional image.

As more root systems were excavated, it became clear that these two-year old pines growing in well tilled soil did not possess the complexity of branching and bending found in older trees on more variable sites. A root system could be adequately characterised by the number and length of its roots. These quantities were measured directly in the field.

4.3.3 Radial distribution

The root systems of 40 three year old trees were winched from the soil, after some loosening around the base of the tree (Figure 1.5). The roots, both lateral and vertical, broke off between 0.1 and 1 m from the base of the trunk. While this approach did not allow total root length to be recorded, it did allow the radial distribution of roots to be assessed. The root systems were cut from the rest of the tree and placed upside down in the centre of a circle marked in 16 segments. The number of lateral roots in each segment was counted, as was the number of vertical roots (angle of declination greater than 45°). While root systems are normally only divided into 4 segments for this type of analysis, 16 segments were used to insure that the resolution would be sufficient to reveal any clumping distribution present.

4.3.4 Inter-tree root contact

Further excavations were made when isolated groups of trees were observed yellowing and dying over a period of 3–4 months. The isolation and row-wise progression of the phenomena suggested a root disease may have been responsible. A back hoe was used to dig a 1.5 m deep trench 1 m from an affected row. Excavation by hand failed to reveal any obvious fungal lesions on the roots. However, it was possible to observe the extent of inter-tree root contact.

Figure 4.6 show first order laterals, confined within a 0.2 m A horizon (top layer), crossing over in a number of places. In many cases the contact was close enough to distort one or both roots. Figure 4.7 (a) and (b) show how the rip line further enhanced inter-tree root contact. At a depth of 0.3 m the rip line remained as a soil-free tunnel of 50 mm diameter running between trees. First order laterals from one tree were channelled through the centre of root system of the next. As the vertical components of the second tree developed, the two root systems became tightly interwoven (Figure 4.7 (b)).

4.4 Sources of data – Summary

Most parameters have been derived from an iterative fitting process in which approximate values have been refined by repeatedly comparing model output with field observations and adjusting input parameters. For example, given that the average length of first order lateral roots at 16 years is known to be l m from root excavation work, the growth rate is assumes to be $l/16$ m a year. It is unlikely that all orders of roots, or even all roots within an order, have a constant extension rate over their lifespan, but Kuiper and Coutss [36] suggest that a linear function is not unreasonable for major first order laterals, which are the most important in terms of inter-tree contact. The result of this assumption is then checked using information from 8 year and 25 year old trees. The convergence of model output with field observations was assessed both by visual comparison and by measures of root distribution such as evenness index and proportion of length by depth [8] discussed in chapter 6.

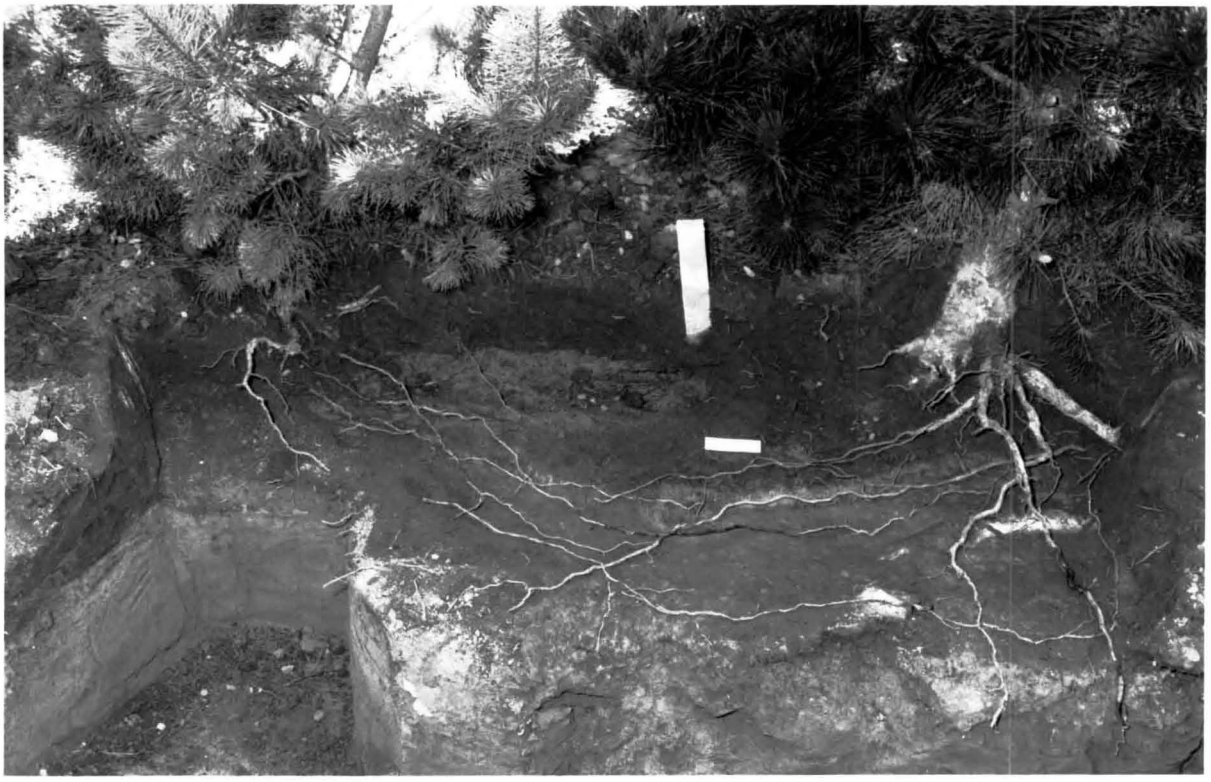


Figure 4.6: Contact between first order laterals of 3 year old *Pinus radiata*.

(a)

Figure 4.7: (a) Adjacent *Pinus radiata* stumps with first order laterals channelled by rip line.

(b)

Figure 4.7: (b) Close up of left tree in (a) showing close root contact.

Chapter 5

Implementation

5.1 Programming environment

All code is in C, and can be compiled either under Borland™ C++ version 3 on an IBM™ AT compatible computer, or under Manx™ Aztec C version 5.2a on a CBM™ Amiga computer. The only significant difference between the IBM and Amiga implementations is in maximum simulation size, where the IBM version is limited to a total program plus data size of 640k, while the Amiga version may use as much memory as is available. Although not essential, fast floating point arithmetic is desirable on either platform. The code can also be compiled under the widely available GNU-C (GCC) version 2.4.5 or greater, and so is easily ported to other platforms. The code has been successfully tested under the public domain Linux operating system with the GCC compiler.

All significant pieces of machine dependent code are encapsulated in the module REALGFX, which provides routines for plotting and drawing in a real co-ordinate system. Scaling and mapping are handled by the REALGFX routines, as is translation from three dimensional to two dimensional space. As discussed in section 8.4, a visualisation system such as this was found to be very valuable during the development of this type of simulation.

5.2 Data representation

The basic data structures used include the following:

```
typedef struct point3 {
  double                  /* a point in 3-dimensional space */
    x,
    y,
    z;
} point3;

typedef struct StemData {
  struct StemData *next;                                    /* not used*/
  point3 pnt;                                               /* location */
  struct RootHeader *mainRt;                                /* tap root */
  Trees tree;                                               /* type of tree */
  double u;     /* Uniform(0,1) - overall size etc. */
} StemData;

typedef struct RootHeader {
  point3 pnt;                 /* location */
  double cLen;                /* current length */
  double fLen;                /* length when mature */
  union rtDat {
    RootPtr next;             /* used for memory management */
    RNodePtr rNode1;          /* start of root */
  } rtDat;
  RootPtr parent;             /* parent root */
  double createT;             /* time when created */
  double phaseT;              /* time of next phase advance */
  RootPhase phase;            /* current phase */
} RootHeader;

typedef struct RNodeData {
  struct RNodeData *next;                           /* next node */
  point3 pnt;                                       /* location */
  double rLen;                                      /* position on root */
  RNodeType type; /* type of node */
  union rnDat {
    RootPtr newRoot;                                /* if New_Root node */
    int cur;     /* used during tree creation */
    FungiPtr fungi;                                 /* if Fungi node */
  } rnDat;
} RNodeData;

typedef struct FungiData {
  union {
    double inc;             /* increase fungal level by this amount */
    long tree;              /* ''tree'' number if rhizomorph */
  } fDat;
  Fungis type;              /* type of fungi */
  BOOLEAN parsed;           /* used during updating */
} FungiData;
```

```
typedef struct StatusType {
  int n;                        /* order */
  double
    pLen,
    fLen;
  point3 pnt;                   /* location */
  double an;                    /* angle */
  double az;                    /* azimuth */
  int
    tot,                        /* total / current, used in tree creation */
    cur;
  StemPtr tree;                 /* tree being processed */
  FILE *fh;                     /* file for output if appropriate */
  fungiArray fLevel;            /* level of each fungi */
  fungiArray cLevel;            /* size of lesion for each fungi */
  fungiArray tLevel;            /* level of each fungi to root collar */
  fungiArray fMod;              /* modifier for each fungi */
  RootPhase phase;              /* phase */
} StatusType;

typedef struct {
  StatusType min;   /* minimum status record */
  StatusType max;   /* maximum status record */
} StatFilter;
```

Root systems are stored as a linked list structure. There are three main structures in the list, the stem, root header, and root node. The stem is not significant in terms of the simulation, and merely represents a useful place to store overall details about each tree. Similarly, the root header contains information about the root it represents, this includes details such as total length of the root when mature, current length of root, and possibly parameters for fungal and physiological condition. The root node contains the spatial information, as each node records a fixed point in three dimensional space.

The creation and processing of this type of linked list is handled by a recursive approach:

```
PerformProcess(Root Header)
      define A as point at which root starts
      define B as first node on root's node list
  [1] process root segment between A and B
      is B a ''New Root'' node?
      if so PerformProcess(Root Header pointed to by B)
      define A as point at which B occurs
      define B as next node on root's node list
      does B exist?
      if so loop back to [1] above
EndOfProcess.
```

The "process" that is applied to the length of root between points A and B varies, it could, for example, be testing the line segment for intersection with a cylinder defined to represent a core sample taken in the field.

5.3 Parameter functions

As described in section 5.7, statistical distributions used to model various aspects of the root system are encoded as C functions that are referenced by a four dimensional array of pointers (references) to functions (a functional unit of program code). The C idiom for variable numbers of function parameters, `stdargs.h`, is used to allow the function `numb(Tree, Order, Feature, Attribute,...)` to return a value sampled from whichever distribution is required. For example, `numb(Pine25, 2, Branch, Number)` (four parameters) will yield the number of branches on a second order root, whereas `numb(Pine25, 3, Fungi, Length, Armillaria)` (five parameters) will return the length of an armillaria lesion on a third order root.

5.4 Construction of the root system

5.4.1 Vector notation

A point in three-dimensional space are denoted as $\vec{P}$; the x, y and z coordinates are referred to as P_x, P_y and P_z. The line segment $\vec{SE}$ runs between the points $\vec{S}$ and $\vec{E}$, whose x, y and z coordinates are S_x, S_y, S_z and E_x, E_y, E_z respectively.

Generation of the simulated root system is recursive, the only important variable being passed down through the recursion being the Status record defined in section 3.3.1. The essential steps are as follows:

1. Allocate a root header and define its final (mature) length.

   `numb(Root,Length,Status)`

2. Initialise the list of nodes on this root by adding the `Root End` node.

3. Determine how many bends this root has, and add these `Root Bend` type nodes to the list of nodes on this root in order of occurrence, location in 3 space as yet undefined, but position along length of root ("root position") established.

   `numb(Bends,Number,Status)`

   `numb(Bends,Position,Status)`

4. Determine how many branches this root has, and add these `New Root` type nodes to the list of nodes on this root in order of occurrence, location as yet undefined.

   `numb(Branch,Number,Status)`

   `numb(Branch,Position,Status)`

5. With the list of nodes now complete and in order, proceed down the list calculating the position of each node based on the current Angle and Azimuth, as recorded in Status and the point occupied by the previous node (In the case of the first node, these are defined by the position of the root header and the supplied Status variable). After each node is positioned the current Angle and Azimuth are adjusted according to the relevant distributions.

   `numb(Bends,Angle,Status)`

   `numb(Bends,Azimuth,Status)`

   `numb(Branch,Angle,Status)`

   `numb(Branch,Azimuth,Status)`

   To calculate the position of each node the following equations are used:

$$E_x = S_x + l_1 \cos(\theta) \quad (5.1)$$
$$E_y = S_y + l_1 \sin(\theta) \quad (5.2)$$
$$E_z = S_z + l \sin(\phi) \quad (5.3)$$

where

$\vec{S}$ = a known point, usually the start of the root or the position of the previous node

l = the distance from $\vec{S}$ to the node being positioned

ϕ = the angle of the root's path approaching this node

θ = the azimuth of the root's path approaching this node
l_1 = abs($l\cos(\phi)$)

The starting position of each root is known either from the position of the `New Root` node on its parent root, or in the case of the 0'th order tap root, from the position specified as an input parameter representing the location of the tree.

6. When a `New Root` type node is encountered, the procedure calls itself with the current Status parameter modified according to the `Branch Vector` distributions for this order.

   `numb(Branch_Vector,Angle,Status)`

   `numb(Branch_Vector,Azimuth,Status)`

7. This recursion continues until a root with zero branches is encountered. In effect `numb(Branch,Number,Status)` must always be zero when the "n" attribute of `Status`, the order of the root for which the parameter is being requested, is N, the highest order of root which exists for the currently defined tree.

5.5 Voxel Lattice

As a root propagates through the area of space occupied by roots from another tree, tests must be made for intersections between the "invading" root (possibly a healthy seedling root) and roots in the "resident" system (possibly an infected root from the previous crop). To save computing time a "voxel lattice" is used to allow testing of only the part of the resident root system (data structure) in the volume of soil in the immediate vicinity of the invading root [7]. Consider the green root in Figure 5.1. To determine which roots it lies nearby, a computationally intensive algorithm could be used to calculate its exact separation from every other root in the simulation. However, if each voxel (cube) represents a list of roots passing through the space the cube encloses, roots nearby the green root can be located by scanning the lists associated with the grey and yellow voxels (the voxels through which the green root passes). As the red root also passes through the yellow voxel, it would be considered to be nearby the green root. The red root's exact separation from the green root can be calculated more precisely.

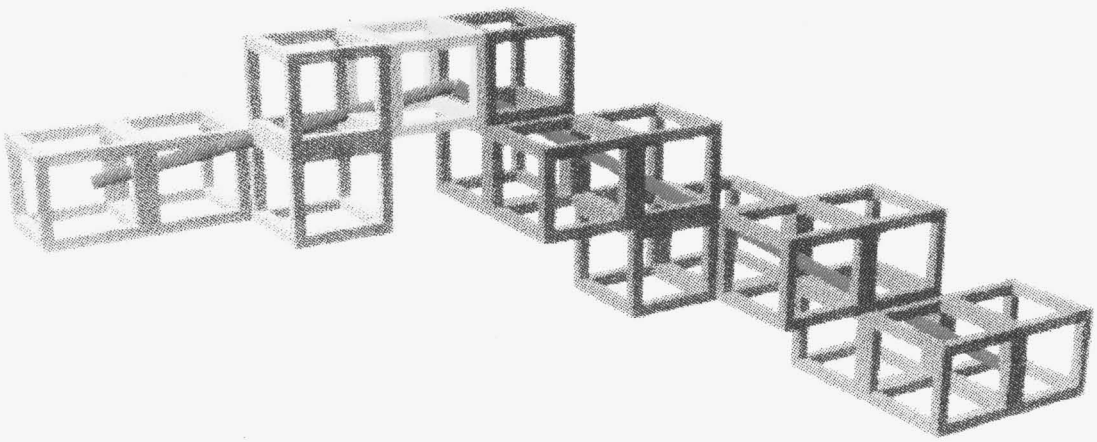


Figure 5.1: Root intersection detection by voxel lattice.

If the root system is completely enclosed in a rectangular prism with a "left front bottom" coordinate of $\vec{V}$ (the voxel lattice origin) and a width, depth and height of w, d, h, then a W voxel wide by D voxel deep by H voxel high lattice would be constructed such that the X, Y, Z element of the lattice would be a list of all the roots contained in or passing through the space with a "left front bottom" coordinate of

$$V_x + X\frac{w}{W}, V_y + Y\frac{d}{D}, V_z + Z\frac{h}{H}$$

and a width, depth and height of

$$\frac{w}{W}, \frac{d}{D}, \frac{h}{H}.$$

In Figure 5.2, a $5 \times 5 \times 4$ ($W \times D \times H$) voxel lattice is shown, with the element $4, 3, 3$ (X, Y, Z) shaded.

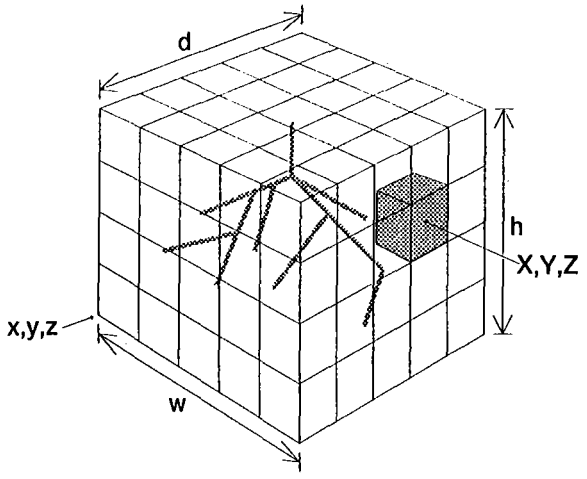


Figure 5.2: A voxel lattice.

Given a point $\vec{P}$, the voxel lattice element containing this point can be found.

$$X = \frac{P_x - V_x}{w/W} \tag{5.4}$$

$$Y = \frac{P_y - V_y}{d/D} \tag{5.5}$$

$$Z = \frac{P_z - V_z}{h/H} \tag{5.6}$$

Given a line segment $\vec{SE}$ representing a infected section of a root from a decaying root system, to check for intersection or close approach between the infected root section and the roots of a developing seedling, it is necessary to determine which voxels the line segment passes through (see Figure 5.1).

First the points defining the beginning and end of the line are checked using Equation 5.4, 5.5, 5.6, then transitions across x, y and z divisions of the lattice are found. For example in the case of x transitions P_x is calculated as follows:

$$P_x = S_x + d \times \text{ceil}((S_X - x)/d) \tag{5.7}$$

where

$\vec{S}$ = the point on the end of the line segment with the lowest x value S_X

$\vec{E}$ = the other end of the line segment

d = the size of each x division $(\frac{w}{W})$

$ceil(a)$ = the lowest integer greater than a

The plane $x = P_x$ is the first division in the lattice encountered moving from $\vec{S}$ to $\vec{E}$.

While P_x is less than E_x, the point of intersection of the line segment with the plane $x = P_x$ is calculated by:

$$P_y = S_y + (E_y - S_y)(t - S_x)/(E_x - S_x) \tag{5.8}$$

$$P_z = S_z + (E_z - S_z)(t - S_x)/(E_x - S_x) \tag{5.9}$$

Each root listed as occupying the voxel containing $\vec{P}$ (Equation 5.4, 5.5, 5.6) is checked for intersection or close approach to the line segment $\vec{SE}$. Finally d is added to P_x until P_x is greater than E_x, then the process is repeated for the y and z divisions of the lattice. Passing

pointers to routines allows the line analysis code to be re-used for adding, removing or locating roots in the lattice.

An internode in a voxel may be very close to an internode in another voxel without passing through, or appearing on the list of, the latter voxel. So to locate all roots near an internode, the voxels adjacent to each voxel the internode intersects should also be checked. The number of adjacent voxels which are included is determined by the minimum voxel dimension (width, depth and height) and the maximum distance over which an internode can interact with nearby objects. For example, an internode passing through voxel X, Y, Z can be considered to be "close" to the roots listed in all voxels from

$$X - \frac{r}{w/W} \, , \; Y - \frac{r}{d/D} \, , \; Z - \frac{r}{h/H}$$

to

$$X + \frac{r}{w/W} \, , \; Y + \frac{r}{d/D} \, , \; Z + \frac{r}{h/H}$$

where

$$r = \text{maximum interaction distance}$$

After the voxel lattice has been used to create a list of internodes in the vicinity of an infected or otherwise noteworthy internode, more computationally intensive routines are used to calculate precise separation distances (section 5.9).

5.6 Analysis of the generated root system

5.6.1 Profile wall analysis

The simulation allows profile wall analysis (see section 6.1.2), with the definition of an arbitrary vertical plane. As elsewhere the root system is parsed recursively, and each line segment, or internode, is checked for intersection with the plane.

Given that a line segment $\vec{SE}$ is the internode under consideration, and the trench wall is the vertical plane whose top left corner is $\vec{T}$ and whose bottom right corner is $\vec{B}$ then:

$$t_1 = \frac{(E_y - S_y)}{(E_x - S_x)} \tag{5.10}$$

$$t_2 = \frac{(B_y - T_y)}{(B_x - T_x)} \tag{5.11}$$

If $t_1 - t_2 = 0$ the internode runs parallel to the trench plane.

To find $\vec{P}$, the point of intersection of the internode with the plane the following equations are used:

$$P_x = \frac{T_y - S_y + S_x t_1 - T_x t_2}{t_1 - t_2} \tag{5.12}$$

$$P_y = S_y + (P_x - S_x)t_1 \tag{5.13}$$

$$P_z = \frac{S_z + (P_x - S_x)(E_z - S_z)}{E_x - S_x} \tag{5.14}$$

If $\vec{P}$ is on both the line segment $\vec{SE}$ and the line segment $\vec{TB}$, when both these lines and the point have been projected into the plane $z = 0$ (Figure 5.3), an internode / trench-wall intersection has been found. If the imaginary grid has C columns and R rows, the column and row are given by Equations 5.15 and 5.16 respectively.

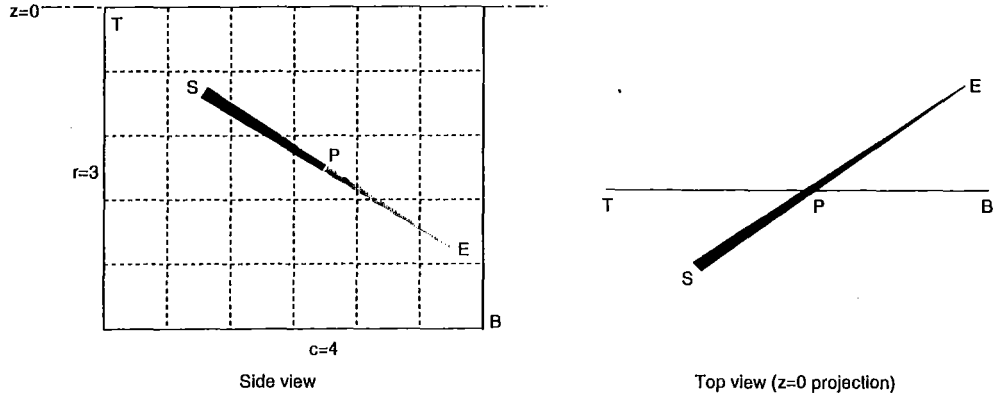


Figure 5.3: Internode intersection with trench wall.

$$c = \sqrt{\frac{(T_x - P_x)^2 + (T_y - P_y)^2}{(B_x - T_x)^2 + (B_y - T_y)^2}} \times C \tag{5.15}$$

$$r = (T_z - P_z)/(T_z - B_z) \times R \tag{5.16}$$

Finally if r and c are in the specified range of rows and columns, the element o, r, c of a three dimensional array (where o is the order of the root) is incremented.

5.6.2 Core sample analysis

The core cylinder (see section 6.1.3) is assumed to be vertical, and is defined by a co-ordinate for the center of the top, a radius, and a length. The sampled volume may be sub-divided into a specified number of layers, segments, and concentric shells. Figure 5.4 shows a cylinder divided into eight segments, with three depth sections and two shells.

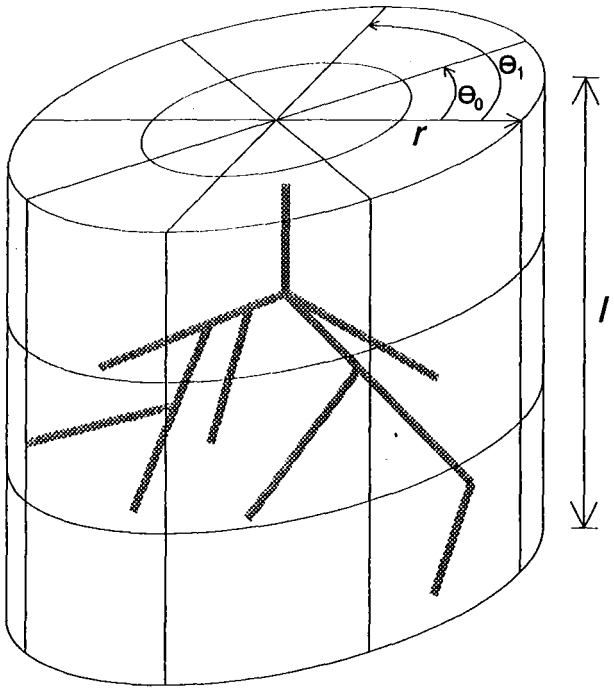


Figure 5.4: A soil core sample.

To find the portion of each internode that lies within the cylinder, two points must be found from

a possible set of eight (Figure 5.5).

Given that an internode under consideration is the line segment $\vec{SE}$, that the center-top of the cylinder is at $\vec{C}$, and that the length and radius of the cylinder are l and r, then the eight points are given by:

$\vec{P_1}$ $\vec{S}$, the beginning of the line segment (internode)

$\vec{P_2}$ $\vec{E}$, the end of the line segment (internode)

$\vec{P_3}$ the intersection of SE with the plane $z = C_z$

$$P_{3x} = S_x + \frac{(E_x - S_x)(C_z - S_z)}{(E_z - S_z)} \tag{5.17}$$

$$P_{3y} = S_y + \frac{(E_y - S_y)(C_z - S_z)}{(E_z - S_z)} \tag{5.18}$$

$$P_{3z} = C_z \tag{5.19}$$

$\vec{P_4}$ the intersection of SE with the plane $z = C_z - l$

$$P_{4x} = S_x + \frac{(E_x - S_x)(C_z - l - S_z)}{(E_z - S_z)} \tag{5.20}$$

$$P_{4y} = S_y + \frac{(E_y - S_y)(C_z - l - S_z)}{(E_z - S_z)} \tag{5.21}$$

$$P_{4z} = C_z - l \tag{5.22}$$

$\vec{P_5}$ the first intersection of SE with the cylinder wall

$$P_{5x} = \frac{-B + \sqrt{(B^2 - 4AC)}}{2A} \tag{5.23}$$

$$P_{5y} = S_y + (P_{5x} - S_x)t_1 \tag{5.24}$$

$$P_{5z} = S_z + \frac{(P_{5x} - S_x)(E_z - S_z)}{(E_x - S_x)} \tag{5.25}$$

where

$$t_1 = \frac{(E_y - S_y)}{(E_x - S_x)} \tag{5.26}$$

$$A = t_1^2 + 1 \tag{5.27}$$

$$B = -2(C_x + t_1(-S_y + t_1 S_x + C_y)) \tag{5.28}$$

$$C = C_x^2 + S_y^2 + 2(-S_y t_1 S_x - C_y S_y + C_y t_1 S_x) + (t_1 S_x)^2 + C_y^2 - r^2 \tag{5.29}$$

$\vec{P_6}$ the second intersection of SE with the cylinder wall

$$P_{6x} = \frac{-B - \sqrt{(B^2 - 4AC)}}{2A} \tag{5.30}$$

$$P_{6y} = S_y + (P_{6x} - S_x)t_1 \tag{5.31}$$

$$P_{6z} = S_z + \frac{(P_{6x} - S_x)(E_z - S_z)}{(E_x - S_x)} \tag{5.32}$$

$\vec{P_7}$ the intersection of SE with the first vertical plane defining the current segment (see profile wall calculation)

$\vec{P_8}$ the intersection of SE with the second vertical plane defining the current segment (see profile wall calculation)

For a point $\vec{P}$ to be one of the two from the set of eight it must fit the following criteria:

1. $\vec{P}$ lies on the line segment $\vec{S}\vec{E}$
2. $\sqrt{(P_x - C_x)^2 + (P_y - C_y)^2} \leq r$
   i.e. distance from $\vec{P}$ to the axis of the cylinder (a vertical line through $\vec{C}$)
3. $\theta_0 \leq \theta \leq \theta_1$
   θ is the azimuth of the line from $\vec{C}$ to $\vec{P}$
   θ_0 and θ_1 are the azimuths of the boundaries of the current segment (Figure 5.4)
4. $C_z - l \leq P_z \leq C_z$

The concentric shells are accounted for by difference between cylinders of different radii. In Figure 5.5 the two points selected would be P_7, an intersection with a segment wall, and P_2, one of the line segment's ends.

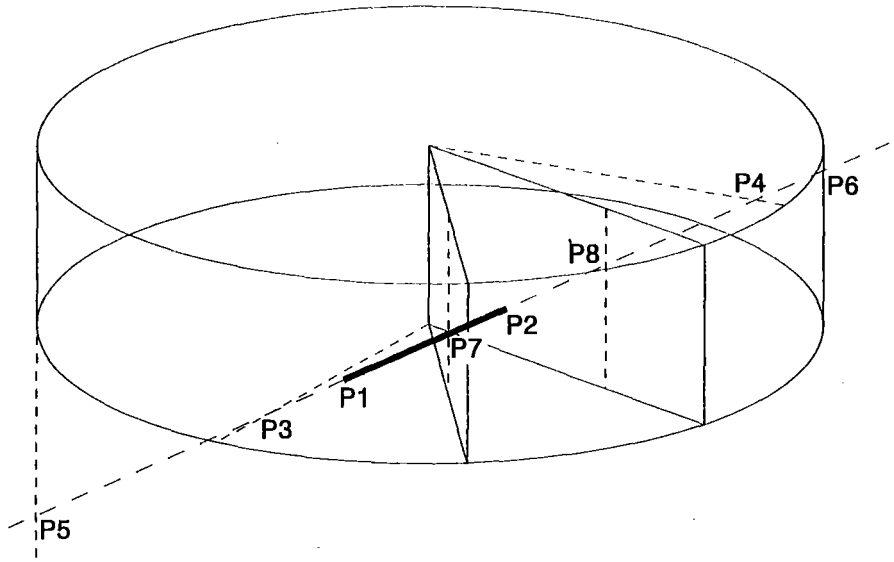


Figure 5.5: Intersection of line segment with cylinder, see text for labeled points.

5.7 Representation of tree statistics

5.7.1 Human readable form

A file of the general form given below was chosen:

```
TreeID,Order,Feature,Attribute:
    any number of lines of C code
    identified by indentation
    should assign the relevant value to 'x'
```

For example,

```
Pine,2,Root,Length:
    x = U(0.5,2);

Pine,2,Branch,Number:
          x = 3;
Pine,2,Branch,Position:
          x = U(0,1);

Pine,TreeDefault,Fungi,Elongation:
          x = 0.005 * yearEffect * timeStep;
          if (st->pnt.z > -0.1) x = 2 * x;
```

declares that the length of 2nd order roots of trees of type "Pine" will be uniformly distributed between 0.5 and 2 m. These roots will branch 3 times, and the positions of those branches will be uniformly distributed along the roots length, U(0,1). Also, the elongation rate of fungal lesions on roots of all orders will be 5 mm a day, adjusted for seasonal effects by the externally defined variable "yearEffect" and scaled up by the number of days in one iteration, "timeStep". Finally, if the fungus occurs in the top 10 cm of soil, the elongation rate will be doubled.

5.7.2 Computer interpretation

Initially the hierarchy of Tree / Order / Feature / Attribute was implemented as a series of nested `switch()` statements. This was difficult to maintain, and the large size of the resulting

code segment prevented compilation on some systems. The human readable form described above (section 5.7.1) was developed, and a small program written to translate this into a valid C structure. For example,

```
Pine,2,Branch,Number:
    int min;
    GET(int,min);
    x = (min > 3) ?  min :  3;
Pine,2,Branch,Position:
    x = U(0,1);
```

would be translated to

```
REAL PROC0000(StatusType *st, va_list vp) {
    int min;
    min = va_arg(vp,int);
    x = (min > 3) ? min : 3;
    return x;
}

REAL PROC0001(StatusType *st, va_list vp) {
    x = U(0,1);
    return x;
}

void initNumb() {

Numb[Pine][2][Branch][Number]=PROC0000;

Numb[Pine][2][Branch][Position]=PROC0001;

}
```

where `Numb` is a 4 dimensional array of pointers to functions, `Pine`, `Branch`, `Number` and `Position` are elements of enumerated types, and `st` is a pointer to pointer to the `StatusType` (section 3.3.1) reflecting the status of the root that the function is being called for. The `GET` macro and `va_list`/`va_arg` system is used to handle additional dimensions in the parameter matrix that affect only some attributes; for example, for the elongation attribute of the fungal feature of a given order of root on a given type of tree, the type of fungus must also be specified.

5.8 Ground topography

The height of the ground surface at any coordinate (x, y) is defined by the function `ground(x,y)`. The first time the function is called, it attempts to open a file "`ground.rs`", which defines the ground height for the points of a grid of arbitrary origin, extent and resolution (Appendix A). A ground surface height of 0 is assumed for any point outside the area covered by the grid. Any point $\vec{P}$ falling within a square (or rectangle) of the grid is treated as lying on a triangular plane segment whose vertices are the three closest grid points. For example, in figure 5.6 $\vec{P}$ lies on the triangle $\vec{A}\vec{B}\vec{C}$. Only the triangles $\vec{A}\vec{B}\vec{C}$ and $\vec{B}\vec{C}\vec{A}'$ are considered.

To simplify the calculation the following re-mappings are made (z is the required value; ground height at $\vec{P}$, P_x, P_y, P_z are the x, y and z coordinates of $\vec{P}$):

$$\vec{P} \rightarrow \left(\frac{P_x - C_x}{B_x - C_x}, \frac{P_y - C_y}{B_y - C_y}, z \right)$$

$$\vec{C} \rightarrow (0, 0, C_z)$$

$$\vec{B} \rightarrow (1, 1, B_z)$$

$$\vec{A} \rightarrow \begin{cases} (1, 0, A'_z) \ , & P_x \geq P_y \\ (0, 1, A_z) \ , & P_x < P_y \end{cases}$$

Note that these re-mappings do not assume the grid is square. If $\vec{P}$ lies on the plane $\vec{A}\vec{B}\vec{C}$ then a line from $\vec{P}$ to any of the points $\vec{A}$, $\vec{B}$ or $\vec{C}$ must be orthogonal to the normal of the plane $\vec{A}\vec{B}\vec{C}$. This orthogonality occurs when (taking $\vec{A}$ as an example)

$$((\vec{A} - \vec{C}) \times (\vec{B} - \vec{C})) \cdot (\vec{P} - \vec{A}) = 0$$

expanding and re-arranging gives

$$z = -\frac{(P_x - A_x)(A_y(B_z - C_z) - (A_z - C_z)) + (P_y - A_y)(A_x(B_z - C_z) - (A_z - C_z))}{A_x - A_y} + A_z$$

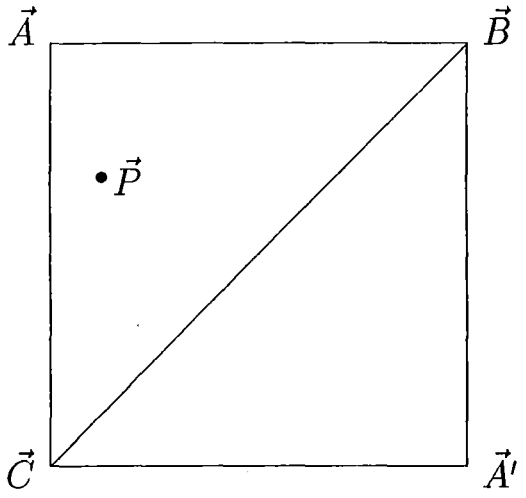


Figure 5.6: A point's position on the ground topography grid.

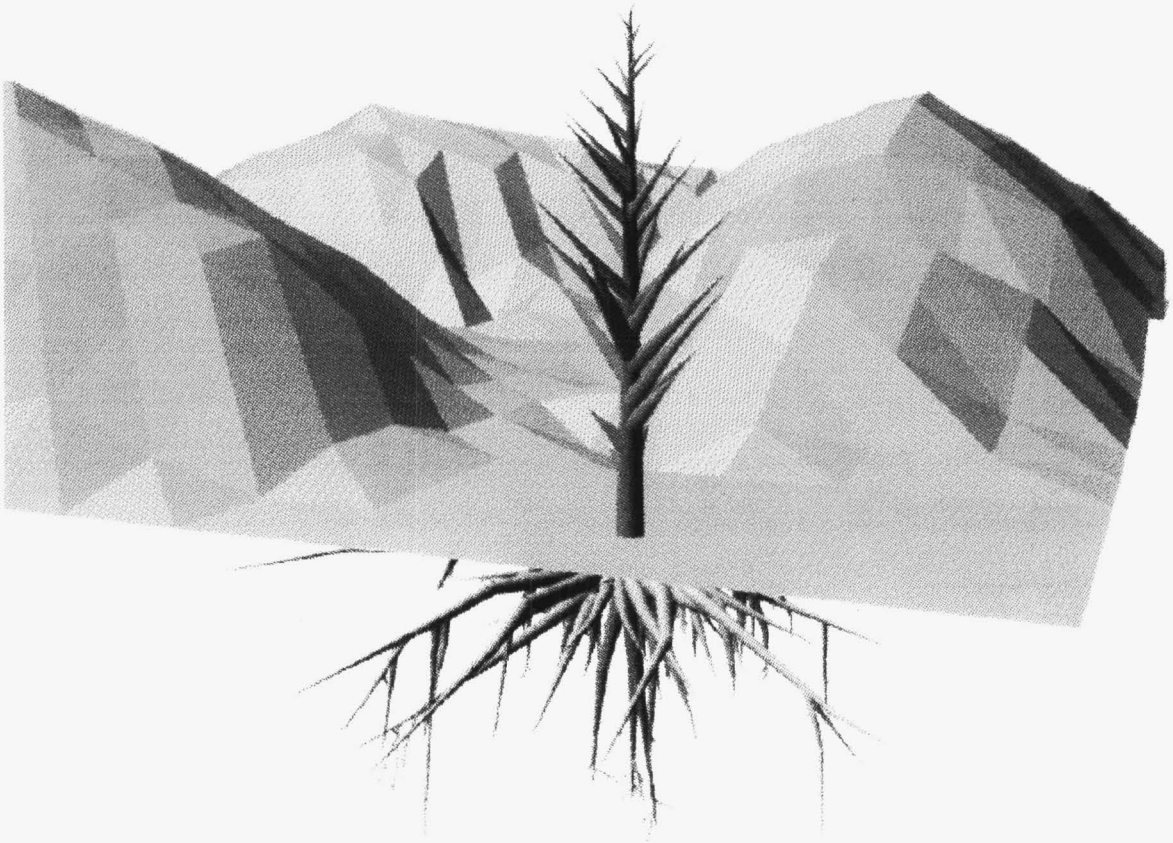


Figure 5.7: A raytrace illustrating the simulation's representation of mounds of soil via. ground topography routines.

5.8.1 Digitising

As described in section 4.1, root system maps, in plan and elevation, were used for the iterative fitting of many of the parameters required by the root architecture submodel. These were digitised using an extension of the simulation software. After initial experimentation with a video frame grabber, a A4 flat bed scanner was used to create single colour "bitmap" images from photocopy reductions of the originals. The simple and widely used PBM, or "Portable Bit Map" format was used, and the scans were made at 300 dpi (dots per inch). In some cases it was necessary to reduce the resolution to 150 dpi for ease of handling.

The simulation software developed was extended to allow interactive editing of a root system using a mouse. The options available include addition of a child root to a selected parent root, removal of a selected root and its children, addition or repositioning of nodes on a root, and cutting of roots at a specified position. Increased precision in the positioning of nodes and new roots is achieved by the use of a cursor that can "slide" up and down an existing root. Zooming and panning are also available, as the root editing features are an extension of the root system viewing features of the simulation software. Editing is only two-dimensional, as the drawings are only two-dimensional, but a transition to three-dimensions would not be excessively difficult, given that the underlying viewing routines already support three dimensions.

To begin entry of the root system drawings, the PBM image files were loaded and displayed, to allow the operator to indicate two points with known coordinates (Figure 5.8 (a)). All drawings were marked with vertical and horizontal scales. With this information the image was scaled appropriately, and used as a backdrop for the mouse controlled root editing process described above. Figure 5.8 illustrates a typical series of root entry steps.

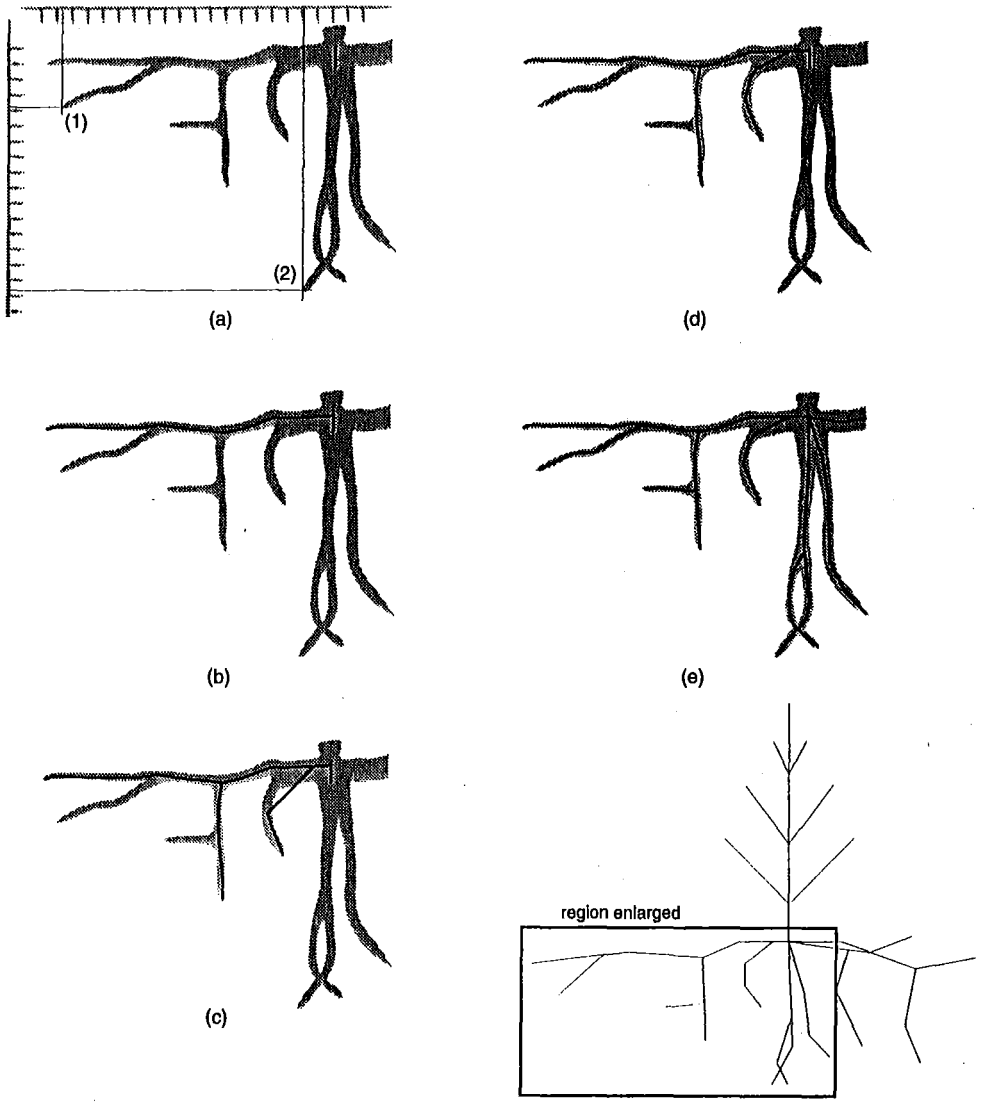


Figure 5.8: Entry of root system from digitised images. (a) Two points selected with known coordinates determine the automatic positioning of a 0th order "stub", (b) a mouse is used to enter the shape of a 1st order root, (c) second order roots are added after selecting their parent root and origin, (d) nodes are added to improve fit to background image, (e) all roots entered.

5.9 Line segment within a sphere

To determine the the position of the line segment $s'\vec{e}'$, $s'\vec{e}'$ being the part of the line segment $\vec{se}$ within a sphere of centre $\vec{p}$ and radius r, the following relationships were used:

$$
\begin{aligned}
(\vec{x}-\vec{p})\cdot(\vec{x}-\vec{p}) &= r^2 \\
\vec{x} &= \vec{s}+\vec{b}t \\
\vec{x} &= \text{a point on the line} \\
\vec{b} &= \vec{e}-\vec{s} \\
t &= \text{an unknown scalar} \\
At^2+Bt+C &= 0 \\
A &= \vec{b}\cdot\vec{b} \\
B &= 2\vec{b}\cdot(\vec{s}-\vec{p}) \\
C &= (\vec{s}-\vec{p})\cdot(\vec{s}-\vec{p})-r^2
\end{aligned}
$$

The two solutions for t give the points of intersection of the line defined by $\vec{s}$ and $\vec{e}$, if either $\vec{s}$ and/or $\vec{e}$ are within r of $\vec{p}$ they are used instead of the points $\vec{s}+\vec{b}t_1$ and $\vec{s}+\vec{b}t_2$.

5.10 Shortest line connecting two line segments.

In a three dimensional system, the shortest route between two line segments (internodes) lies along one of nine paths. The route will run from either end or some intermediate point on one line segment to either end or some intermediate point on the other line segment (Figure 5.9).

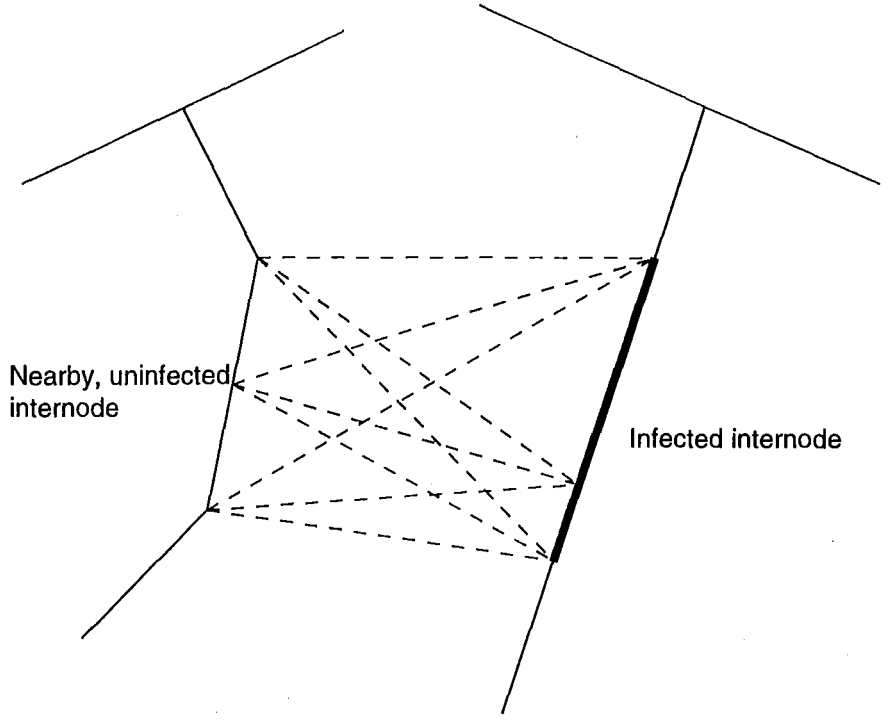


Figure 5.9: Nine paths, one of which will contain the shortest route between two line segments.

Given that the line segments S_1 and S_2 run between the points $\vec{p_1}$ to $\vec{p_2}$ (S_1) and $\vec{p_3}$ to $\vec{p_4}$ (S_2) (Figure 5.10).

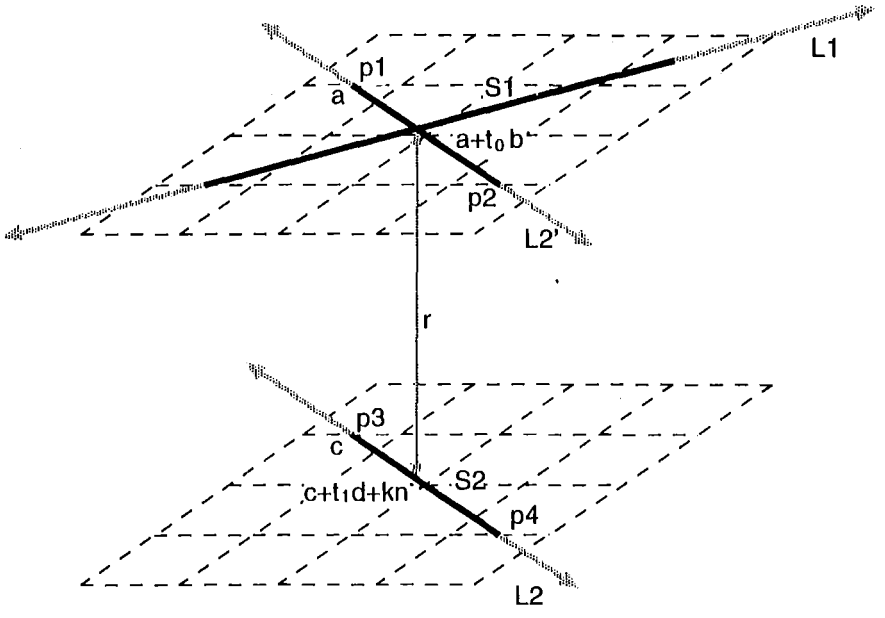


Figure 5.10: Shortest route between two line segments

1. Find the shortest distance between the lines (L_1 and L_2) that the line segments are part of. The lines are considered to lie in parallel planes whose shared normal ($\vec{n}$) is calculated, then the separation of the planes (r) is determined. $\vec{a}$, $\vec{b}$, $\vec{c}$ and $\vec{d}$ are defined for convenience.

$$L_1 = \vec{a} + t\vec{b} \tag{5.33}$$

$$\vec{a} = \vec{p_1} \tag{5.34}$$

$$\vec{b} = \vec{p_2} - \vec{p_1} \tag{5.35}$$

$$L_2 = \vec{c} + t\vec{d} \tag{5.36}$$

$$\vec{c} = \vec{p_3} \tag{5.37}$$

$$\vec{d} = \vec{p_4} - \vec{p_3} \tag{5.38}$$

$$\vec{n} = \vec{b} \times \vec{d} \tag{5.39}$$

$$r = \frac{\vec{n} \cdot \vec{a} - \vec{n} \cdot \vec{c}}{||\vec{n}||} \tag{5.40}$$

2. Project L_2 onto the plane of L_1 to force intersection, yielding L_2'. So two scalars, t and k, exist

$$L_2' = \vec{c} + t\vec{d} + k\vec{n} \tag{5.41}$$

such that

$$||k\vec{n}|| = r \tag{5.42}$$

Rearranging,

$$k = \pm\sqrt{\frac{r^2}{\vec{n} \cdot \vec{n}}} \tag{5.43}$$

the sign of k should be selected to match the sign of r, after which r can be set to $|r|$.

Substituting L_2' for $\vec{x}$ in the symmetric equation for L_1

$$\frac{x_1 - a_1}{b_1} = \frac{x_2 - a_2}{b_2} = \frac{x_2 - a_2}{b_2} \tag{5.44}$$

leads to

$$\frac{c_1 + td_1 + kn_1 - a_1}{b1} = \frac{c_2 + td_2 + kn_2 - a_2}{b2} = \frac{c_3 + td_3 + kn_3 - a_3}{b3} \tag{5.45}$$

which yields

$$t = \frac{b_2b_3(c_1 + kn_1 - a_1) + b_1b_3(c_2 + kn_2 - a_2) - b_1b_2(c_3 + kn_3 - a_3)}{b_2b_3d_1 + b_1b_3d_2 - 2b_1b_2d_3} \tag{5.46}$$

this results in t begin undefined for lines parallel to one of the three axis (i.e. when an element of $\vec{b}$ is zero), so one of the three possible sub solutions, eg.

$$t = \frac{b_2(a_1 - c_1 - kn_1) + b_1(c_2 + kn_2 + a_2)}{b_2d_1 - b_1d_2} \tag{5.47}$$

is chosen to avoid a zero denominator.

The point on L_1 closest to L_2, $\vec{x_1}$, is

$$\vec{c} + t\vec{d} + k\vec{n} \tag{5.48}$$

and the point on L_2 closest to L_1, $\vec{x_2}$, is

$$\vec{c} + t\vec{d} \tag{5.49}$$

3. The points $\vec{x_1}$ and $\vec{x_2}$ lie on the lines L_1 and L_2, but not necessarily on the line *segments* LS_1 and LS_2. t_0 and t_1 are calculated for the equations

$$\vec{a} = \vec{p_1} + t_0\vec{b} \tag{5.50}$$

and

$$\vec{c} = \vec{p_3} + t_1\vec{d} \tag{5.51}$$

then if $0 \leq t_0 \leq 1$ and $0 \leq t_1 \leq 1$ the points of closest approach for L_1 and L_2 occur on LS_1 and LS_2 and are used to define the shortest route between the line segments.

otherwise one of the 4 ends must be the closest point to the other line segment, and the distance from each end to the other line segment must be calculated and the shortest selected. For a point $\vec{p}$ (one of the 4 ends) the point ($\vec{o}$) on the line segment $\vec{m}\vec{n}$ nearest to $\vec{p}$ is

$$l = \frac{(\vec{n} - \vec{m}) \cdot (\vec{p} - \vec{m})}{\vec{n} \cdot \vec{n}} \tag{5.52}$$

$$0 \leq l \leq 1 \longrightarrow \vec{o} = \vec{m} + l(\vec{n} - \vec{m}) \tag{5.53}$$

otherwise

$$\sqrt{\vec{p} \cdot \vec{a}} < \sqrt{\vec{p} \cdot \vec{b}} \longrightarrow \vec{o} = \vec{m} \tag{5.54}$$

otherwise

$$\vec{o} = \vec{n} \tag{5.55}$$

and then $\vec{x_1'}$, the point on LS_1 nearest LS_2 is selected accordingly, and likewise for $\vec{x_2'}$, the point on LS_2 nearest LS_1.

5.11 Azimuth from coordinates

The azimuth (compass bearing) A from one node (x_1, y_1) to another (x_2, y_2) is calculated using

$$A(x,y) = \begin{cases} \pi/2 & x = 0, y > 0 \\ -\pi/2 & x = 0, y \leq 0 \\ \arctan(\frac{y}{x}) & x > 0 \\ \arctan(\frac{y}{x}) + \pi & x < 0, y \geq 0 \\ \arctan(\frac{y}{x}) - \pi & x < 0, y < 0 \end{cases} \tag{5.56}$$

where

$$x = x_2 - x_1$$

$$y = y_2 - y_1$$

5.12 Trapezoidal distributions

Given the large number of statistical distributions required for the root-architecture submodel, and the lack of statistically significant data sets for many variables, a quick and effective method for sampling from weakly characterised distributions has obvious value. However the triangular distribution commonly used in such cases cannot represent bi-modal and other more complex distributions without modification. Instead, algorithms were written to sample a "trapezoidal" distribution. A series of trapezoids can be fitted to approximate any bounded distribution, in practice the numerical accuracy of most computer systems allows very wide, flat trapeziums to adequately approximate unbounded distributions as well.

Figure 5.11 shows the use of a trapezoidal distribution to model the initial angle of first order lateral roots. This angle has a bimodal distribution, sharply bounded 0, the soil surface, and $-\pi$, the angle of a vertical root. Most roots are approximately horizontal or approximately vertical, but roots with angles of declination in between do occur. Therefore, simple triangular distributions are not suitable to model this component of root system morphology. The trapezoidal distributions are defined by a series of pairs of numbers representing the "height" and position of the sides of the trapeziums, adjacent trapeziums share a side. A n trapezium distribution can be represented with either a $2 \times (n + 1)$ array or a procedure accepting an undefined number of parameters, using the `stdargs.h` facility of the C programming language. The uniform distribution is sampled twice, once to determine which trapezium the sample is in, and again to determine where the sample lies within the trapezium.

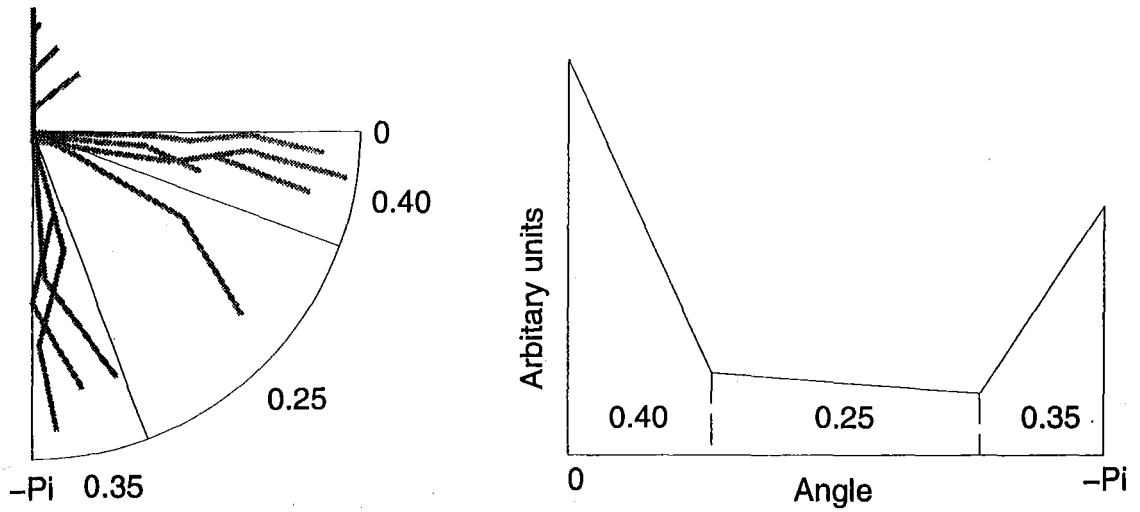


Figure 5.11: A trapezoidal distribution.

Chapter 6

Validation

As this simulation aims to evaluate and develop a methodology as much as it aims to answer specific questions about the *Pinus radiata* pathosystem, validation has been limited. The highly variable system under study and the lack of extensive replicated field data exclude rigorous statistical analysis. However, one sub-model, root architecture, raises particular validation issues, and received consideration.

6.1 Root architecture

The root system morphology of *Pinus radiata* is an example of a complex entity that cannot be adequately characterised by a small number of simple variables. Numerous descriptive indices of plant root form are available, but their application is generally subjective. While this subjectivity does not prevent their use for within trial comparison, it does preclude them from model validation roles. The radial distribution index developed here relies on both field and model data being available in computer readable form to achieve repeatable, objective comparison.

6.1.1 Radial distribution of lateral roots

The uneven radial distribution of lateral roots (horizontal roots radiating outwards from the base of the trunk) is a recognised feature of *Pinus radiata* root architecture. Menzie's lateral root score is a typical example of the subjective indices used to measure such variables [3]. An objective

index of lateral root evenness has been developed [8]. The index divides the root system into N segments (typically 64), then forms a summation of the deviation of the amount of root material in each segment from that expected for a perfectly even distribution ($1/N$th). The index is weighted for different levels of scale, for example an unexpectedly high concentration of root material occurring in one 16th of the root system contributes less to the total score than a similar (proportionally) deviation occurring in one 4th of the system. When ranking the same sets of root maps on (un)evenness, agreement between visual interpretation and the index is excellent.

An objective evenness index can be defined for N segments, numbered $0...N-1$, as follows:

$$\text{Index} = \sum_{c=0}^{(\log_2 N)-1} \sum_{r=0}^{l} \left(\left| \frac{\sum_{i=r}^{r+2^c-1} O_{(i \bmod N)}}{d} - 1 \right| \times w \right) \tag{6.1}$$

where

O_i = $N \times$ the proportion of root mass or length occurring in segment i

l = $N-1$ for a complete circle

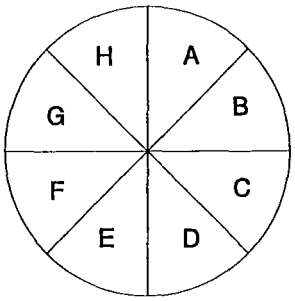
l = $N - 2^c$ for a semicircle or other broken ring of segments

d = 2^c for a completely even distribution, other functions could be used is some uneven but regular distribution was expected

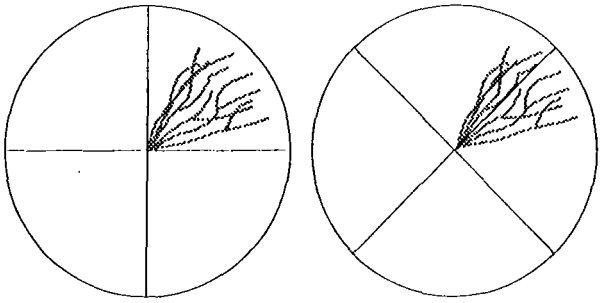
w = 2^c to weight large scale unevenness as being more important than small scale unevenness, this can be adjusted as appropriate

$a \bmod b$ = the remainder of a after integer division by b

Figure 6.1 (a) illustrates how the index avoids the dependence on the initial rotation of the reference frame (Figure 6.1 (b)) that adds an element of subjectivity to similar indexes.



(a) Index is sum of deviation in segments
A, B, C, ... H
plus deviation in all quarters
AB, BC, CD, ... HA
plus deviation in all halves
ABCD, BCDE, ... HABC

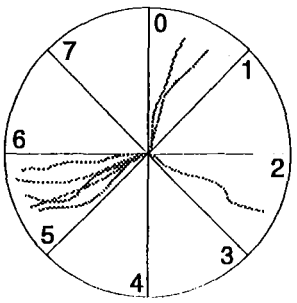


(b) The same tree scores 1/0/0/0 or 0.5/0.5/0/0, depending on segment orientation

Figure 6.1: Overlapping and alignment of segments in "evenness" index.

Example application of the index.

Given the eight root system pictured, the index is calculated as follows:



Segment	O_i
0	2
1	0
2	1
3	0
4	0
5	5
6	0
7	0

$$\mathcal{A} = \sum_{i=r}^{r+2^c-1} O_{(i \bmod N)} \qquad \mathcal{B} = \left| \frac{\mathcal{A}}{d} - 1 \right|$$

	segments c = 0 d = 1 w = 1			quarters c = 1 d = 2 w = 2			halves c = 2 d = 4 w = 4			
r	i	$\mathcal{A}$	$\mathcal{B}$	i	$\mathcal{A}$	$\mathcal{B}$	i	$\mathcal{A}$	$\mathcal{B}$	
0	0	2	1.0	0,1	2	0.0	0,1,2,3	3	0.25	
1	1	0	1.0	1,2	1	0.5	1,2,3,4	1	0.75	
2	2	1	0.0	2,3	1	0.5	2,3,4,5	6	0.50	
3	3	0	1.0	3,4	0	1.0	3,4,5,6	5	0.25	
4	4	0	1.0	4,5	5	1.5	4,5,6,7	5	0.25	
5	5	5	4.0	5,6	5	1.5	5,6,7,0	7	0.75	
6	6	0	1.0	6,7	0	1.0	6,7,0,1	2	0.50	
7	7	0	1.0	7,0	2	0.0	7,0,1,2	3	0.25	
			10.0			6.0			3.50	total
			10.0			12.0			14.00	weighted

Total evenness index 36.

For comparison, Figure 6.2 illustrates some other possible arrangements and their scores. For eight equal roots in eight segments, the maximum score (occurring when all eight roots are in one segment) is 70. The failure to distinguish between the two cases which score 36 in Figure 6.2 is a side effect of using only 8 segments, the index intended for computerised application to precise root maps or image data, where a larger number of segments would be used.

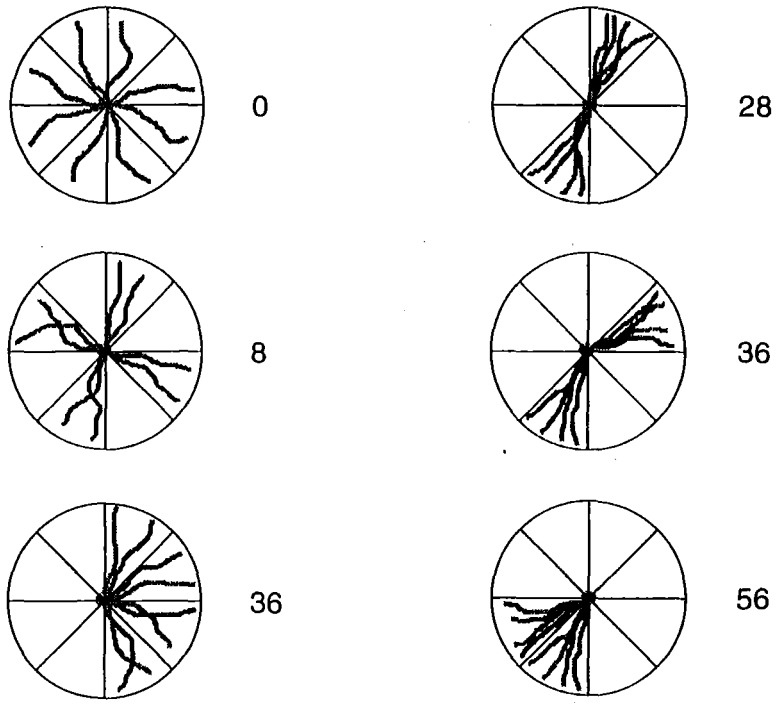


Figure 6.2: Examples of (un)evenness index scores.

6.1.2 Profile walls

A very common form of root system study in the field involves the excavation of a trench, with one smooth vertical wall through the root system. The positions of the intersections of roots with this wall are recorded, usually on an intersections per grid-square basis. In some cases diameters may also be recorded. Root order and direction of travel cannot be directly measured.

Given the abundance of such studies in the literature, it is useful for the simulation software to be able to perform the same type of analysis on the modelled root architecture. The observed and simulated grids of intersection counts can then be compared to assess the model's validity, or how successfully the model has been calibrated to a given plant-species / soil combination. As the 3-dimensional position of each internode is recorded, determining the position of the intersection of each root with an arbitrarily defined vertical plane is relatively simple. Figure 6.3 shows the calculated intersections for a simulated root system.

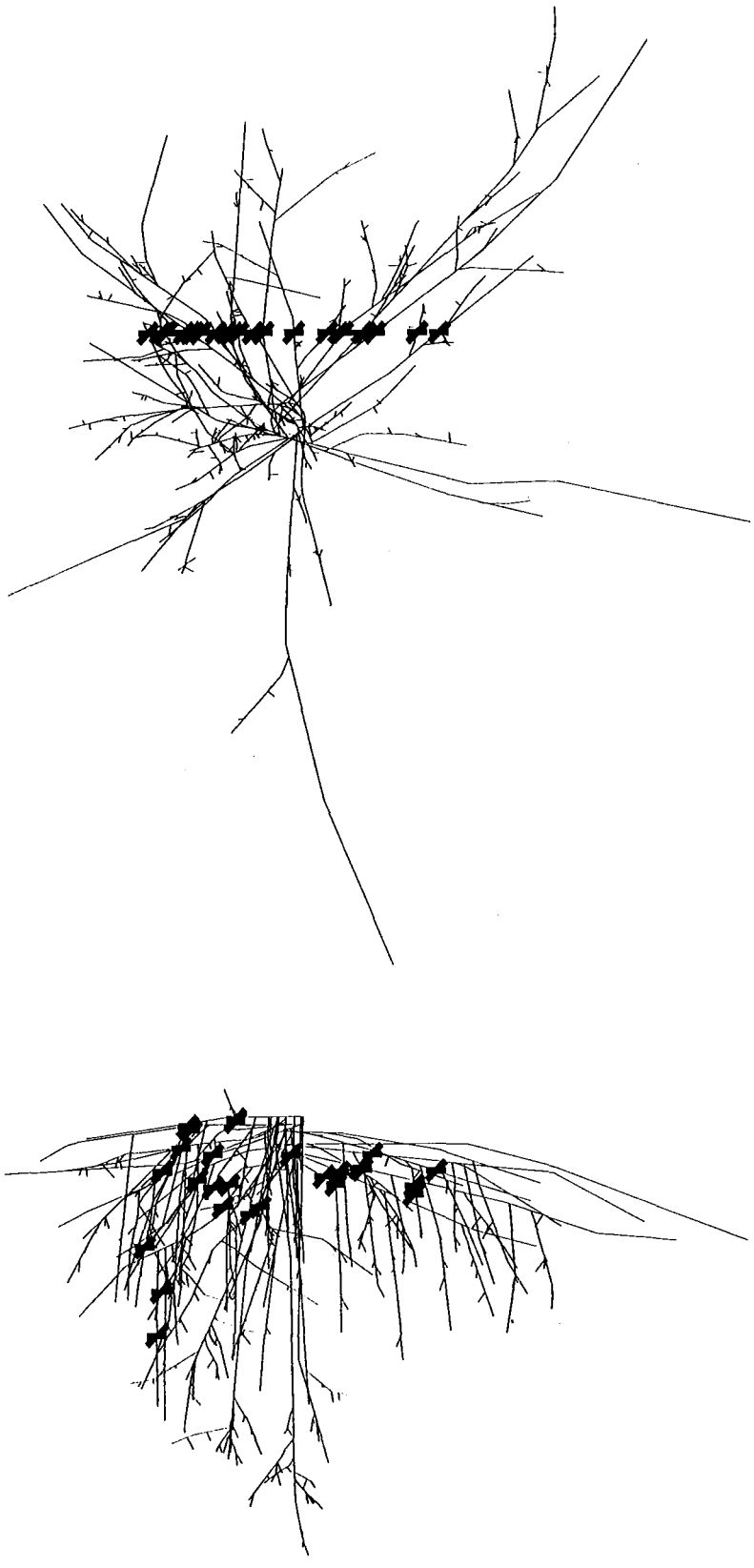


Figure 6.3: Trench intercept, plan and elevation.

6.1.3 Core sampling

Another common root-system field measurement is the taking of core samples, by forcing a sharp edged cylinder down through the soil. This form of analysis may also be performed on the simulated root system. To allow comparison with some existing analysis, the cylinder is further subdivided. It is common practice to divide the extracted core sample into sections from different depths, and sometimes useful to measure root characteristics in segments, and in concentric "shells" of soil.

As the simulated analysis incurs none of the large resource overhead that applies to its field counterpart, the analysis routines required to measure roots in a 10 cm core can be used with an arbitrarily large radius, to allow, for example, layer by layer characterisation of an entire root system. Figure 6.4 shows the calculated intersections for a simulated root system.

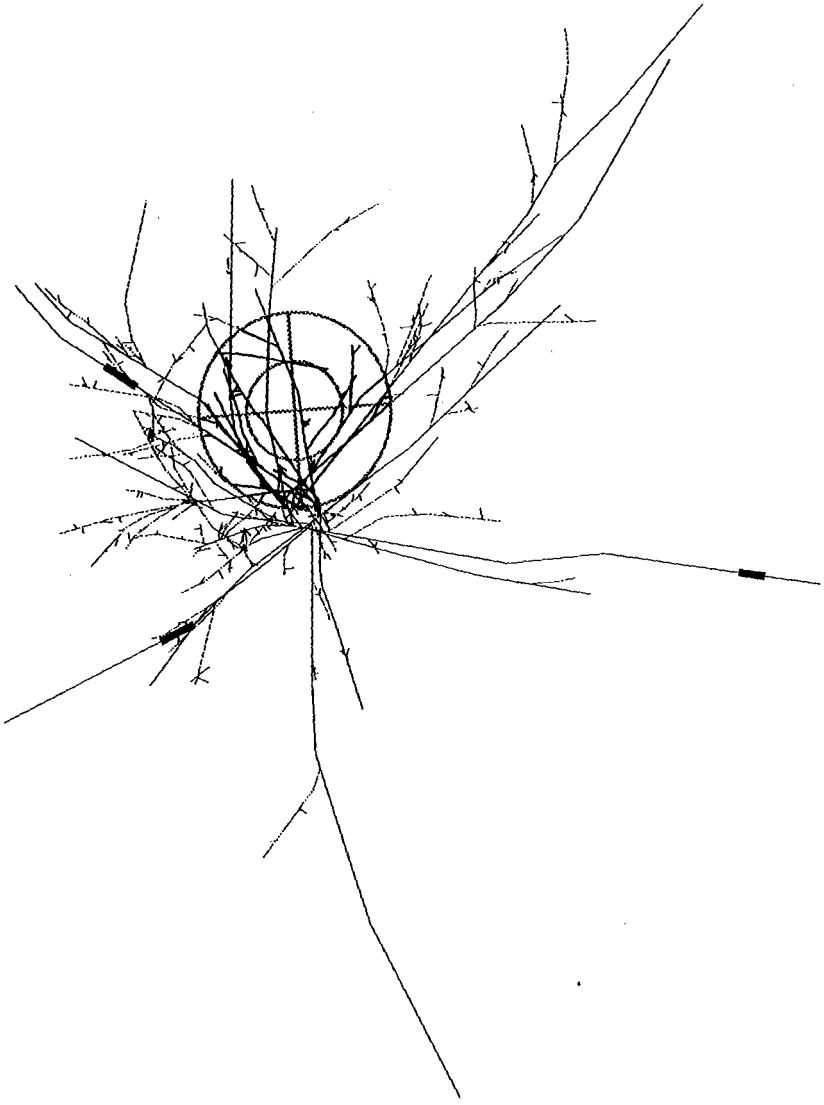


Figure 6.4: Core sample.

Chapter 7

Results

The simulation model developed is quite detailed, and therefore able to be applied to a very wide range of phenomena. Three phenomena of particular importance to the *Pinus radiata* / armillaria pathosystem, disease centre expansion, thinning and biological control application, have been chosen to test the model's implementation. Disease centre expansion has a major effect on the diseases impact on established plantations. Thinning, which kills root systems and so makes large substrate pools suddenly available to armillaria, may have a positive or negative effect on disease impact, as killed root systems may act as either isolating gaps or substrate rich sources of infection. The results can be considered as a successful phase validation of the simulation. They confirm that root system architecture and plant and fungal interactions are modelled with sufficient accuracy to produce the same patterns of behaviour that are observed in the field.

7.1 Disease centre expansion

7.1.1 Observed behaviour

The map of armillaria distribution presented here (Figure 7.1), generously made available by Zeglen [66], illustrate the way in which armillaria is distributed in the field (in this case a North American stand). This map was one of several showing a similar disease distribution. Radial expansion of armillaria from disease centres created by the previous crop is well documented, although interpretation of exactly how the expansion occurs varies [42, 48, 61].

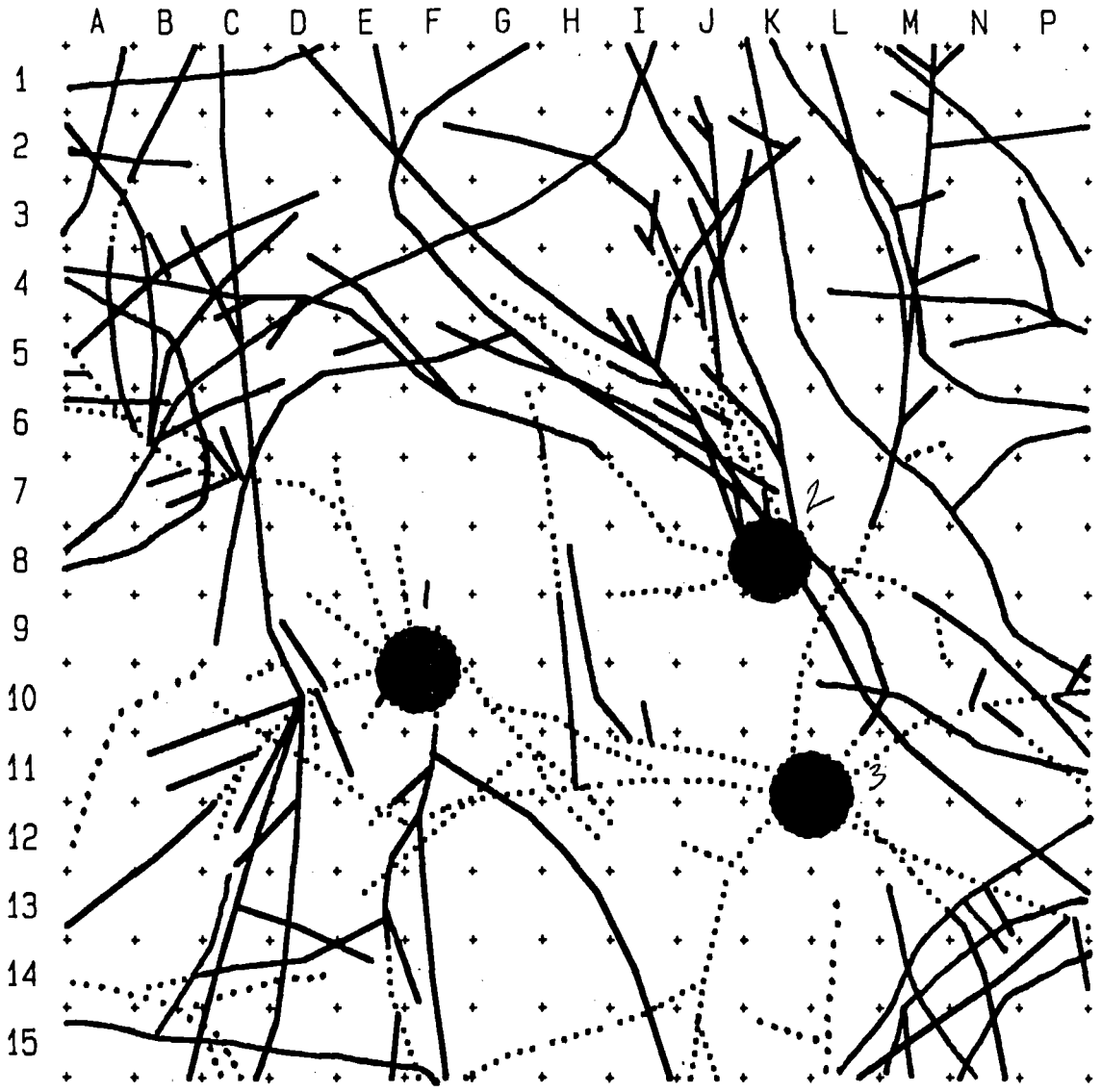


Figure 7.1: Armillaria distribution in a lodgepole pine stand; steady state in a North American forest, field observation. Dotted lines represent infection.

7.1.2 Model response

A simulated four by four tree stand with inter- and intra-row spacing of 2 m was generated. All root systems were advanced to the "Growing" physiological stage. Armillaria lesions were placed on a root system in a corner of the stand, and the root system was advanced to the "Recently Dead" stage (see section 3.5 and Appendix B). All other root systems were free of fungal infection. The initial condition is shown in Figure 7.2 (a). The model was allowed to run for the simulated time of 3500 days, with a 15 day time step. Every 60 days the disease's advance was plotted, and the root system was tested for possible cross infection sites. The time shown on all figures is measured in days.

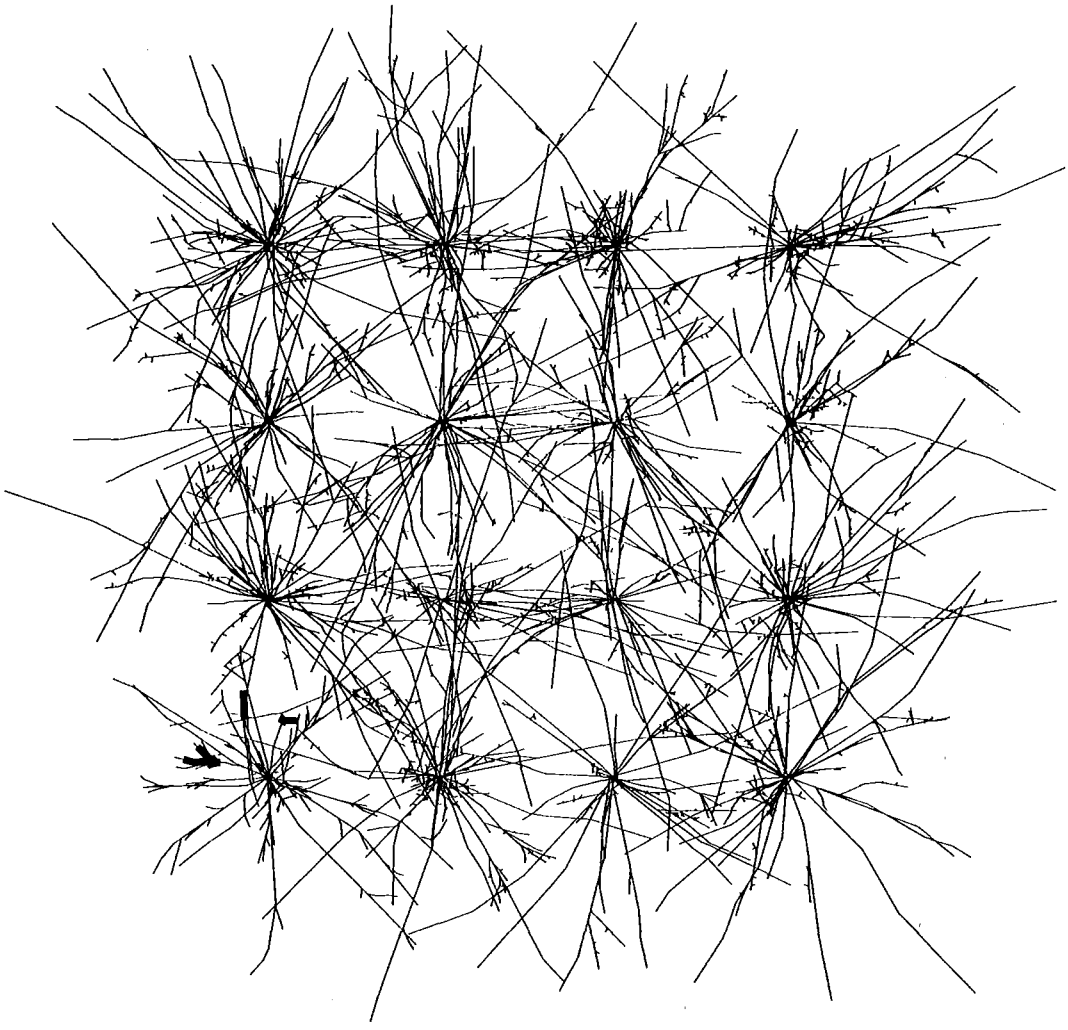


Figure 7.2: (a) Spread of armillaria in a simulated stand. At 60 days – the initial condition, note sections of infected root on tree at bottom left.

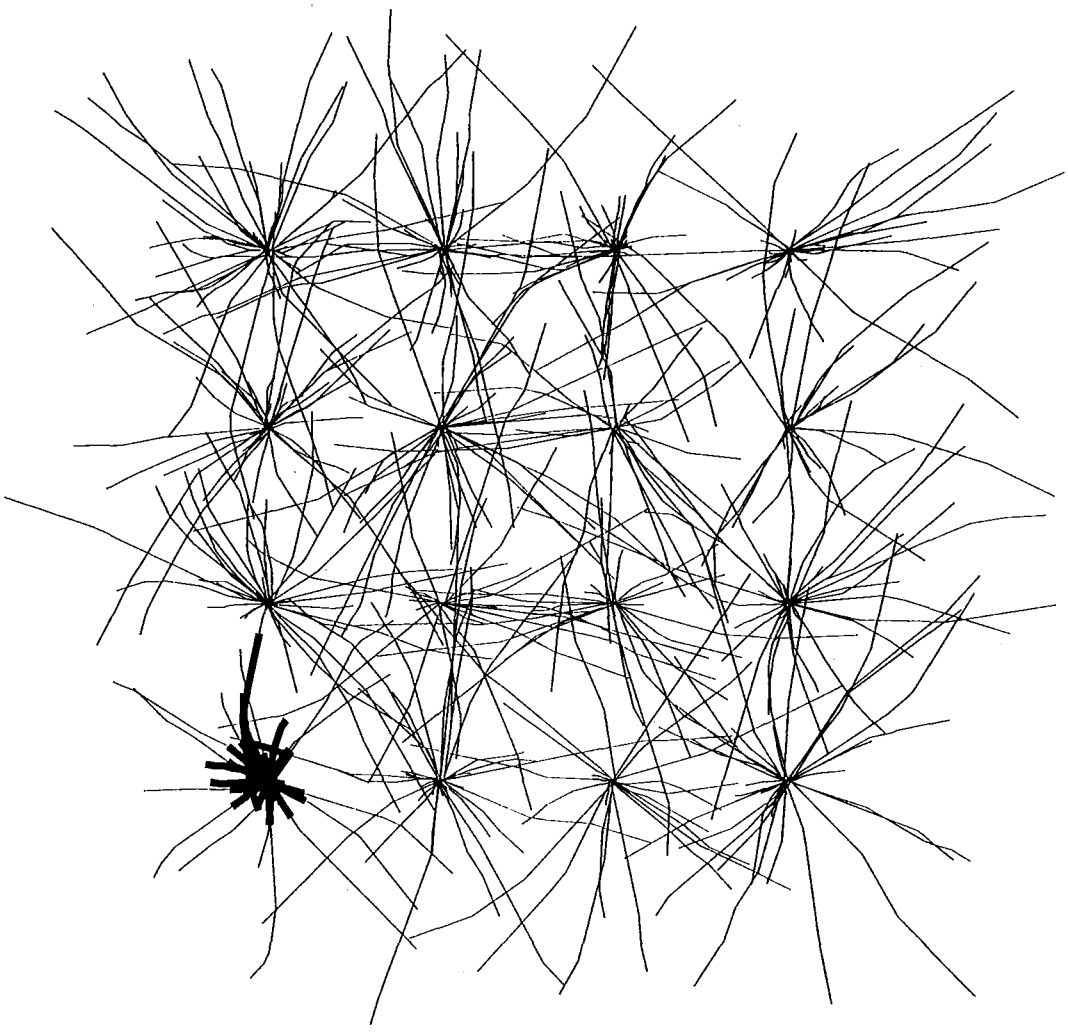


Figure 7.2: (b) Spread of armillaria in a simulated stand. At 600 days – armillaria begins to expand in the dead root system. The synchronisation of fine root death seen here is a side-effect of running the simulation with matured trees, as opposed to trees simulated from the seedling stage.

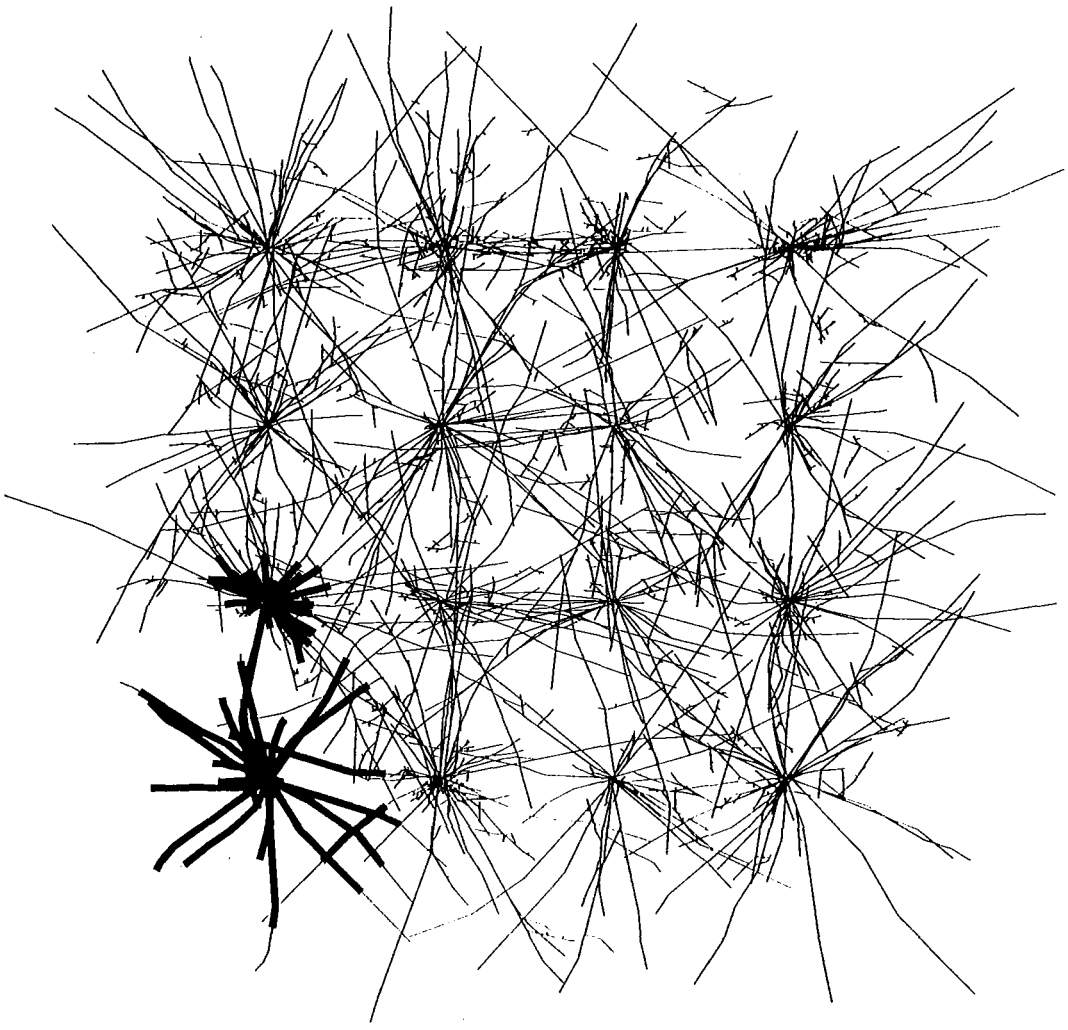


Figure 7.2: (c) Spread of armillaria in a simulated stand. At 1200 days – the root collar is probably an important infection locus, once the infection reaches the collar, spread to the rest of the root system is almost inevitable.

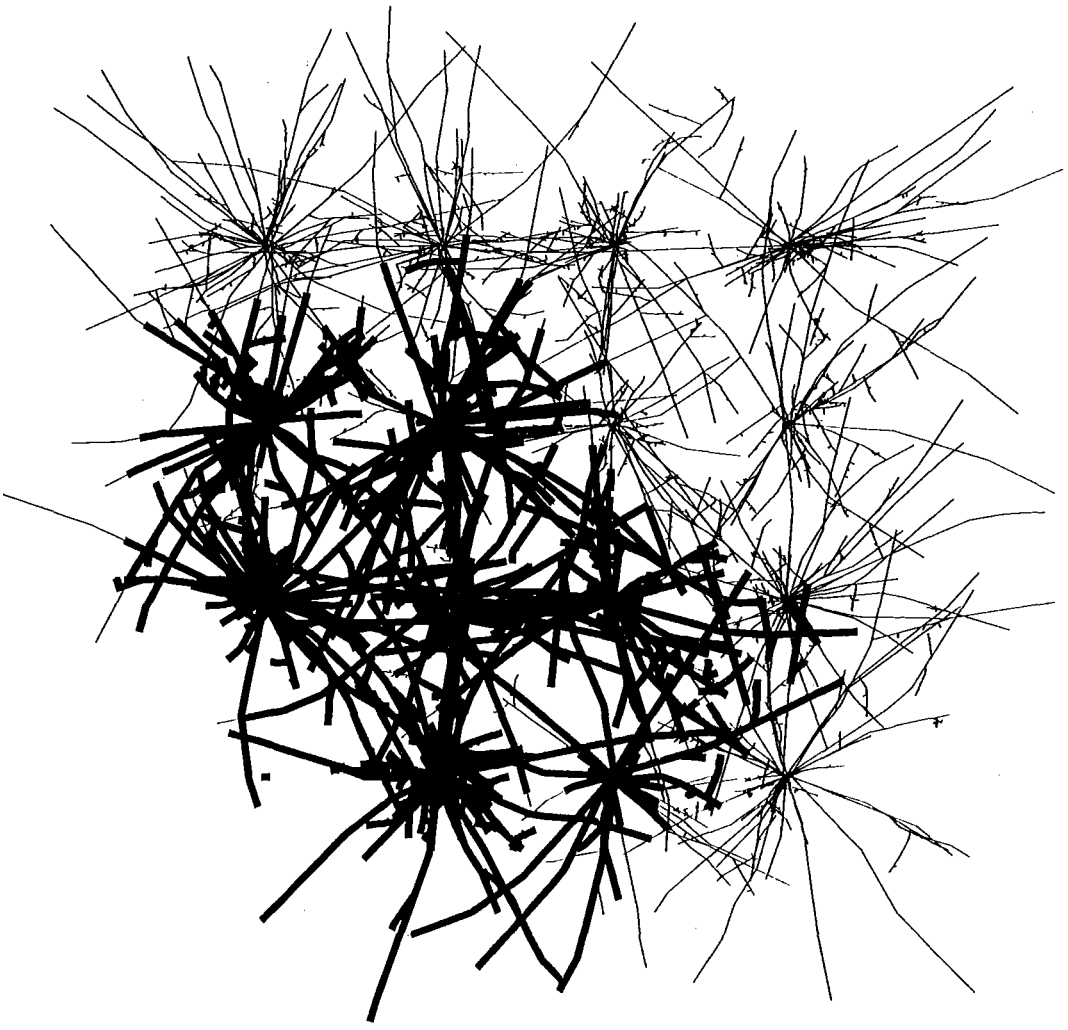


Figure 7.2: (d) Spread of armillaria in a simulated stand. At 3420 days – armillaria spreading outwards in a radial fashion.

7.2 Thinning

7.2.1 Observed behaviour

For the situation illustrated in Figure 7.3, where thinning creates a dead root system between infected and uninfected trees, the impact of thinning is variable. Thinning may cause infection of tree C, by providing a path of lessened resistance, assuming recently killed roots are more susceptible to armillaria than actively growing roots. Thinning may prevent infection of tree C by terminating root extension and eliminating contact between either trees A and B or B and C. Thinning may have no effect on disease spread, if, regardless of thinning, no contact occurs between A and B or B and C, or if A comes into direct contact with C.

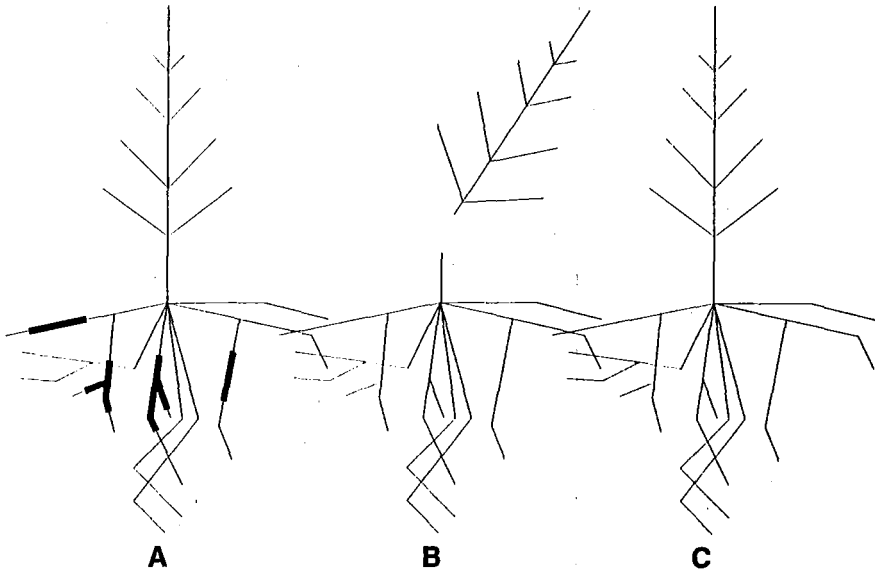


Figure 7.3: Thinning adjacent to a diseased root system.

7.2.2 Model response

A simulated row of three trees at 2 m spacing was generated. To represent a dead root system from the previous crop, the first tree was advanced to the "Recently Dead" physiological stage, and included several armillaria lesions. The middle and last trees were allowed to grow from the seedling state, without initial fungal contamination. The simulation was run twice, first without any intervention, and then with the "thinning" of the middle tree at 1095 days (3 years). Thinning

was modelled by advancing the root system to the "Recently dead" physiological stage.

The results are show in Figure 7.4, (a)-(d). The following extract from the simulation log file for the unthinned run shows that cross infection almost occurred at 1320 days (fungal score 0.99, root score 1.00), but was delayed until the next cross-infection check, when the increased size of the armillaria lesion allowed it to overcome the growing root's resistance.

Log file codes are as follows:

FTOO:	successful inter-root infection
FTO-:	unsuccessful inter-root infection
(inter)	inter-tree event
(intra)	intra-tree event
FTOP:	child to parent fungal spread
FTOC:	parent to child fungal spread
A:	armillaria level on source root
TS:	trichoderma level on source root
TD:	trichoderma level on destination root
FM:	fungal (armillaria) multiplier (section 3.9.1)
F:	fungal score
R:	root score

The number after the physiological phase name indicates the order of the root

```
1320.0: Cross Infect
1320.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:19.89 TS:0.00 TD:0.00 FM:0.10 F:0.99 R:1.00
1320.0: Save Image 0.000000,0.000000,0.000000
        0.000000,0.000000,0.000000 1.000000
1380.0: Cross Infect
1380.0: FTOO : Decayed 1 to Growing 1 (inter),
        A:22.29 TS:0.00 TD:0.00 FM:0.10 F:1.11 R:1.00
```

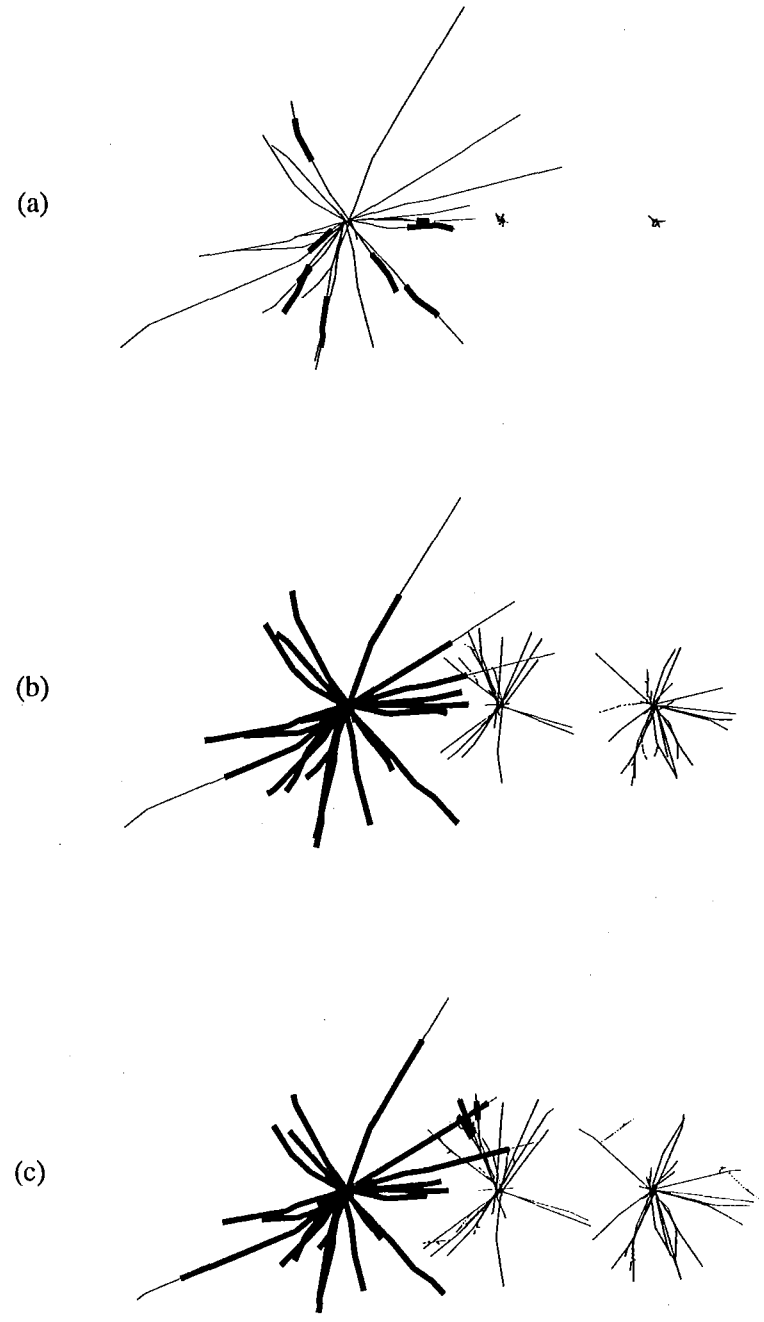
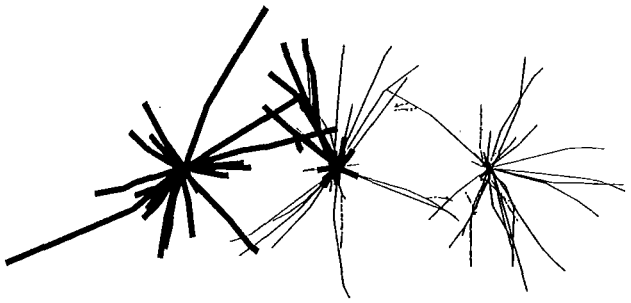


Figure 7.4: Thinning prevents cross infection. (a) At 120 days – the simulation's initial condition. (b) At 1200 days – it appears that cross infection may take place, but in fact the infected root is situated too deeply for contact to occur. (c) At 1500 days – cross infection does occur on day 1380, and is clearly visible by day 1500.

(d)



(e)

(f)

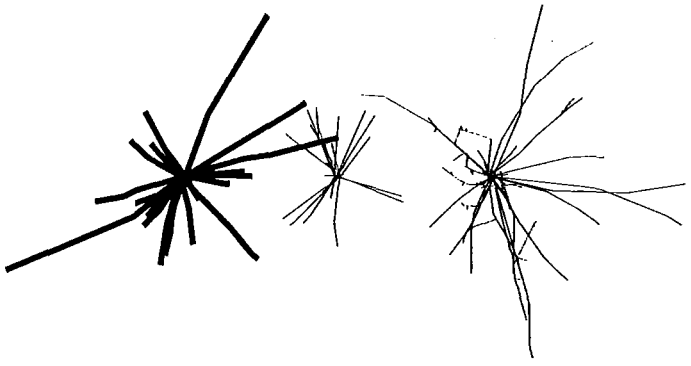


Figure 7.4: Thinning prevents cross infection. (d) At 2040 days – armillaria spreads through the central seedling. (e) At 3000 days – armillaria spreads from one seedling to the next. (f) At 3000 days – with thinning, neither seedling is infected, even if the central seedling had been infected, the infection would not have reached the next seedling.

As described above, thinning may also *cause* cross infection. The following log file extract illustrates how a root which resists cross infection while alive (fungal score F:0.81 < root score R:1.00, fungal multiplier FM:0.10) is infected after thinning at 1825 days (5 years) (F:24.35 > R:1.00, FM:3.00)

```
1800.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:16.23 TS:0.00 TD:0.00 FM:0.10 F:0.81 R:1.00
1800.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:16.23 TS:0.00 TD:0.00 FM:0.10 F:0.81 R:1.00
1800.0: Save Image 0.000000,0.000000,0.000000
        0.000000,0.000000,0.000000 1.000000
1825.0: Mature Tree 1 1 5 0
1860.0: Cross Infect
1860.0: FTOO : Decayed 1 to Recently Dead 1 (inter),
        A:16.23 TS:0.00 TD:0.00 FM:3.00 F:24.35 R:1.00
```

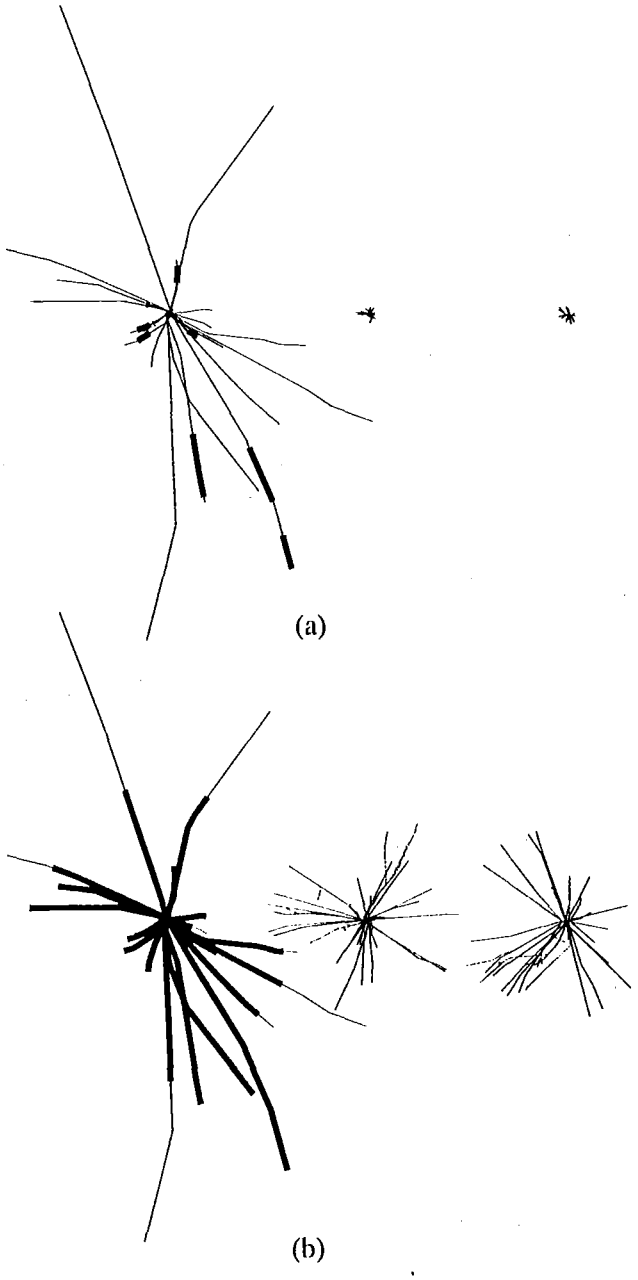
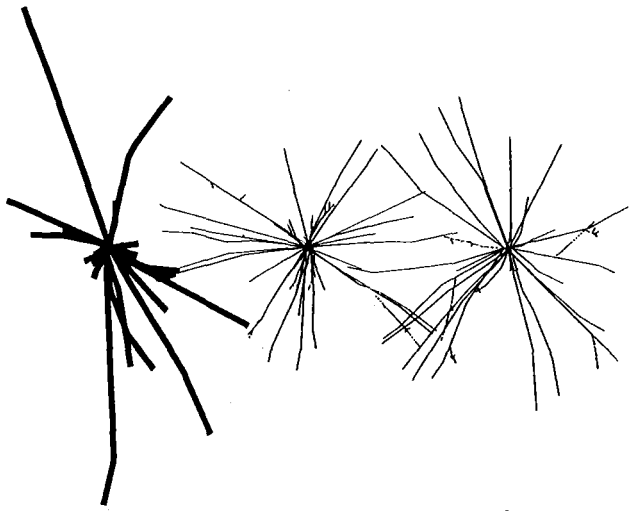
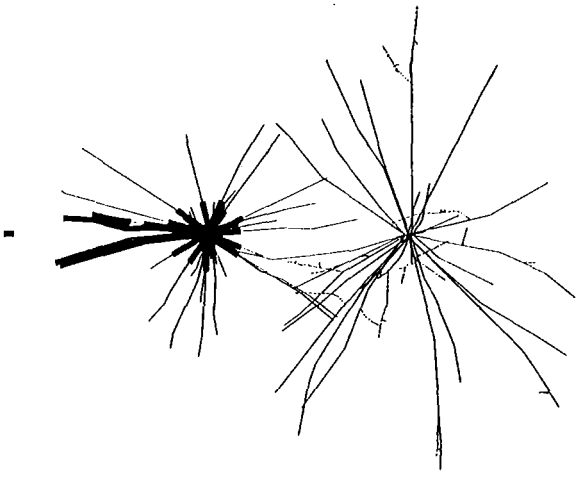


Figure 7.5: Thinning causes cross infection. (a) At 120 days – the initial position. (b) At 1140 days – armillaria spread through the dead root system.

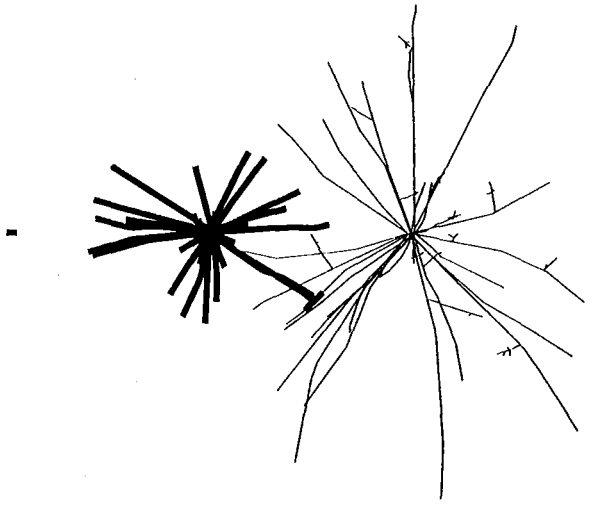


(c)

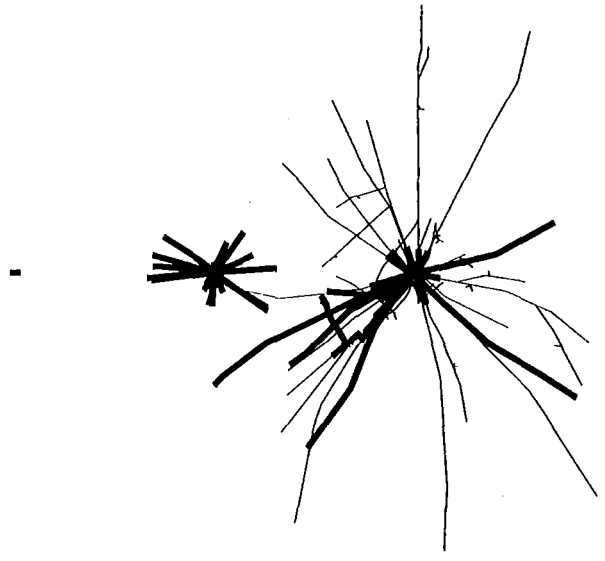


(d)

Figure 7.5: Thinning causes cross infection. (c) At 1920 days – cross infection of the central seedling occurs. The seedling has been made more accessible to armillaria by thinning. (d) At 2760 days – armillaria is well established in the central seedling.



(e)



(f)

Figure 7.5: Thinning causes cross infection. (e) At 3300 days – cross infection of the second seedling occurs. (f) At 3960 days – the second seedling is extensively infected.

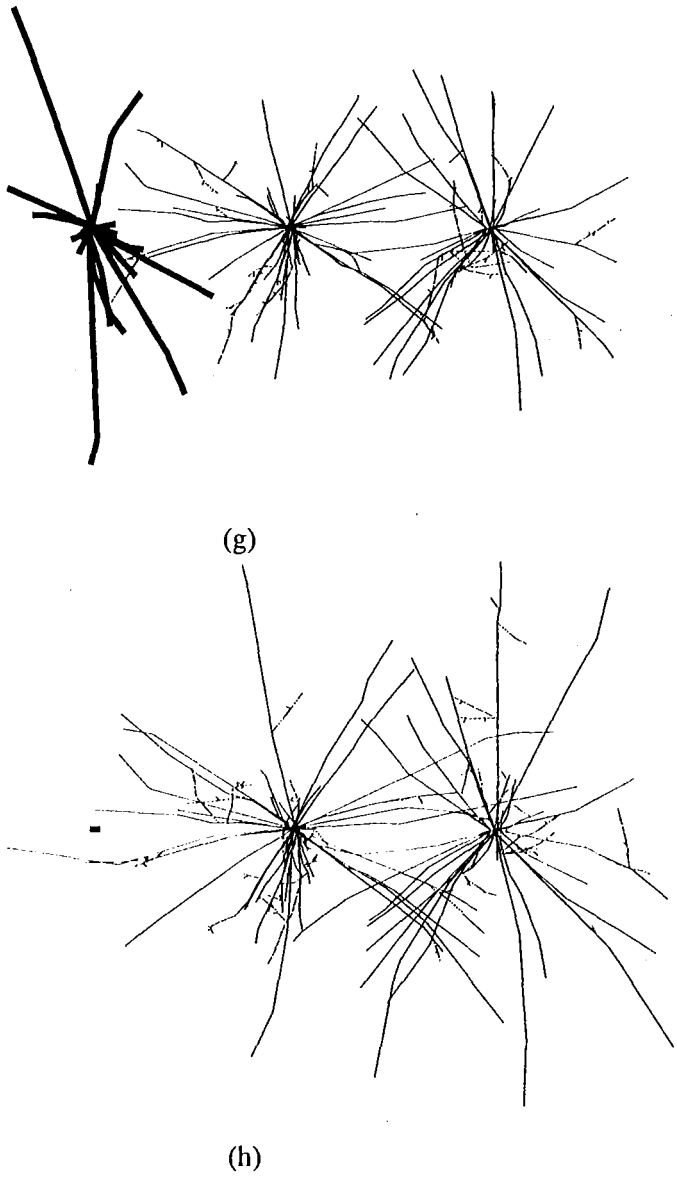


Figure 7.5: Thinning causes cross infection. (g) At 2160 days – the more resistant, non-thinned root system resists cross infection. (h) At 3960 days – both seedlings survive uninfected.

7.3 Biological control application

7.3.1 Observed behaviour

Cutler and Hill [17] (see section sTrichArm) reported significant suppression of armillaria induced mortality in young *Pinus radiata* stands by *Trichoderma hamatum* and *T. virde* preparations applied in the nursery. Most bio-control programmes, in fact most pest control programmes in general, can benefit from optimisation in terms of timing, and in many cases positioning, of the application of the bio-control organism. Optimising there parameters gives the greatest protection for the least cost, and may make the difference between a successful and an ineffective control. This section tests the simulation's ability to model the effects of *Trichoderma spp.* as an antagonist of armillaria.

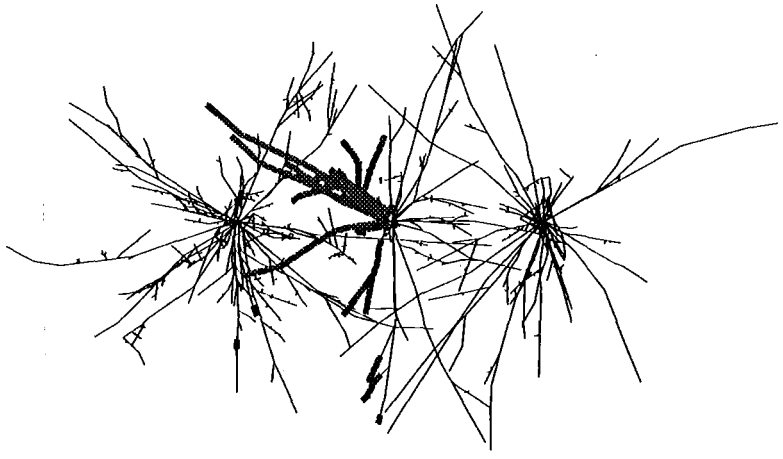
7.3.2 Model response

A simulated row of three trees at 2 m spacing was generated, as described in section 7.2. The simulation was run twice, first without any intervention, and then with the application of "trichoderma" to the soil surrounding the seedling.

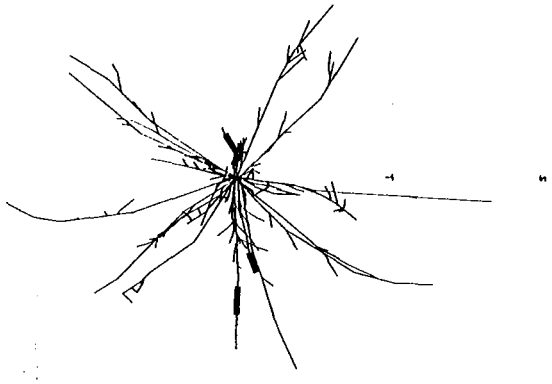
The following log file extracts show an immediately successful cross infection event without trichoderma, and repeatedly unsuccessful cross infection challenges with trichoderma.

```
1740.0: FTOO : Decayed 1 to Growing 1 (inter),
        A:68.12 TS:0.00 TD:0.00 FM:0.10 F:3.41 R:1.00

1740.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:68.12 TS:0.00 TD:13.06 FM:0.10 F:3.41 R:13.06
1740.0: Save Image 0.000000,0.000000,0.000000
        0.000000,0.000000,0.000000 1.000000
1800.0: Cross Infect
1800.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:68.12 TS:0.00 TD:13.58 FM:0.10 F:3.41 R:13.58
1800.0: Save Image 0.000000,0.000000,0.000000
        0.000000,0.000000,0.000000 1.000000
1860.0: Cross Infect
1860.0: FTO- : Decayed 1 to Growing 1 (inter),
        A:68.12 TS:0.00 TD:14.09 FM:0.10 F:3.41 R:14.09
```

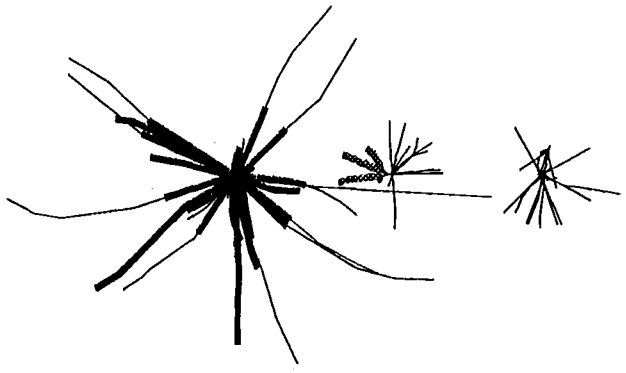


(a)

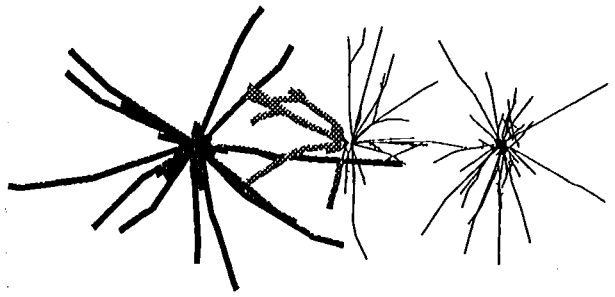


(b)

Figure 7.6: Simulated *Trichoderma spp.* biocontrol prevents cross infection. (a) Initial (pre-run) state of simulation, with root systems displayed in their fully grown state, i.e. the "temporal envelope" for the simulation. Roots growing through soil containing trichoderma are shown in grey. (b) At 60 days – initial position of the root systems.

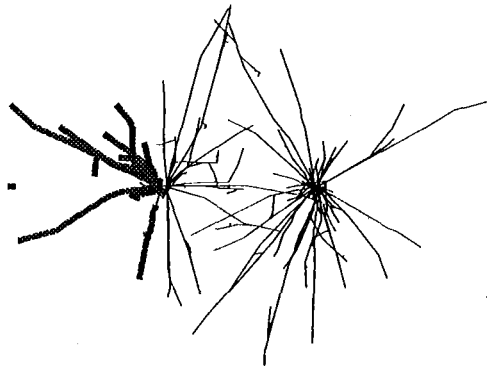


(c)

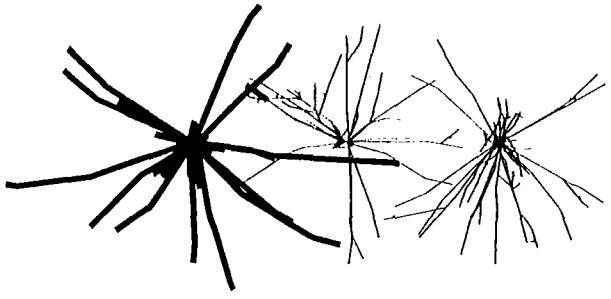


(d)

Figure 7.6: Simulated *Trichoderma spp.* biocontrol prevents cross infection. (c) At 840 days – the central seedling starts to grow into the soil volume containing trichoderma. (d) At 1860 days – seedling and infected root systems overlap, long roots from the infected root system suggest more extensive trichoderma application would afford greater protection.

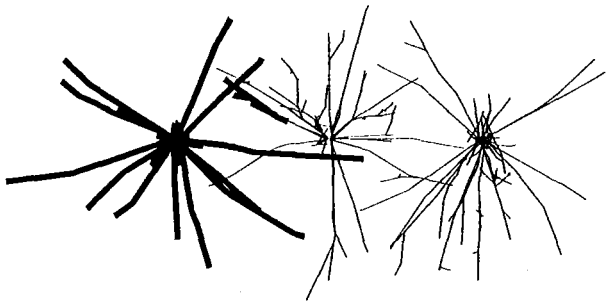


(e)

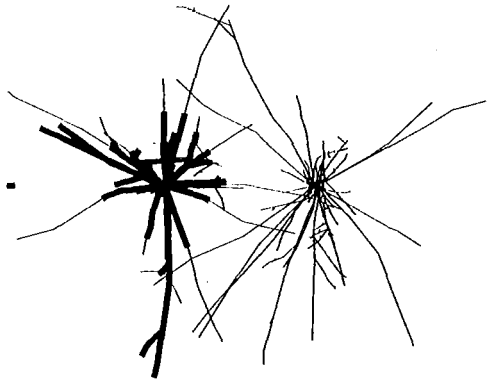


(f)

Figure 7.6: Simulated *Trichoderma spp.* biocontrol prevents cross infection. (e) At 2940 days – the infected root system has decayed completely without causing infection of either seedling. (f) At 1860 days – re-running the simulation without trichoderma leads to cross infection.



(g)



(h)

Figure 7.6: Simulated *Trichoderma spp.* biocontrol prevents cross infection. (g) At 2160 days – even in the absence of trichoderma, the root systems make contact at only one point. This limited contact emphasizes the impact of the positioning of roots. (h) At 2940 days – without the trichoderma barrier armillaria spreads through the central seedling.

Chapter 8

Discussion

8.1 Model response to trial situations

While the results presented in the previous chapter are a confirmation of the potential of the technique developed here to model a plant root microbial ecosystem, they are not the major result of this work. The main contribution is the development of the technique described in chapter 3 and the implementation of that technique to a stage where its functionality can be assessed.

8.1.1 Disease centre expansion

The pattern of armillaria distribution seen in Figure 7.1 is perhaps more typical of the steady state situation thought to exist in native forests in New Zealand, or in North American forests. The disease centre expansion simulation (section 7.1.2 and Figure 7.2) is intended to model spread from what is essentially a point source. This is probably closer to the situation that exists in recently established *Pinus radiata* plantations in New Zealand [30]. Nonetheless, the essential similarity between the North American root maps and the simulation output suggests that the simulations representation of the pathosystem is a realistic one.

The rate of spread is probably excessive, as is the total infection the disease achieves (compare with Figure 7.1). The former is a result of the crude parameterisation for variables such as fungal growth rates. The latter could addressed by more detailed representation of the interaction between root physiology and fungal colonisation (section 9.2). Given these deficiencies, the general pattern seen in the output is certainly that which was intended. As the model specification

itself contains no element dealing with the concept of radial disease centre expansion, this result verifies the functionality of the model's intended mode of action, namely the representation of large scale phenomena by the modelling of small scale mechanisms.

8.1.2 Thinning

As discussed in section 7.2.1, the thinning may have a positive, negative or neutral effect on the spread of disease between any two trees. The results from the thinning simulation runs indicate that the simulation is capable of producing all of these possible outcomes. A useful application of the simulation would be in the optimisation of, for example, the time of thinning for a range of spacings and armillaria densities. This could be achieved simply by recording the effect of thinning for multiple runs of the program with various spacing, timing, and randomisation values. Without thorough parameterisation such results would of course be meaningless.

8.1.3 Biological control application

Here again the result of simulation runs with and without biological control agent application serve more as a verification of the model's ability to represent a phenomena rather than as an output suitable for validation purposes. The system's ability to represent the spatial aspects of biological control are confirmed. Accurate modelling of the outcome of pathogen / control agent encounters will be more easily achieved with the development a more realistic fungal physiology submodel as discussed in section 9.2, although in the case of trichoderma and armillaria the relatively simple case of complete suppression may be all that is required [17].

8.2 Application to the *Pinus radiata* / armillaria pathosystem.

The biological control of armillaria in kiwifruit orchards with *Trichoderma hamatum* and *T. viride* has been a spectacular success [17]. Similarly promising results have been seen in trials of trichoderma in *Pinus radiata* plantations, although the different spatial and temporal scales involved mean that conclusions as to the treatment's effectiveness are as yet incomplete (section 1.3.2). The simulation technique developed here may yet find useful application to the analysis of the *Pinus radiata* / armillaria pathosystem. However, it has taken three years to develop and test

the model, without thorough parameterisation. With a larger development group, algorithm and software development and testing could probably have been completed within one to one and a half years. Parameterisation could have been conducted in parallel to produce a simulation ready for application to field problems within a total of one and a half to two and a half years, approximately.

Such a delay, particularly in the current climate of short term funding, would generally be a serious impediment to the useful application of a simulation to a research project. The usefulness of models and simulations in complimenting field research programmes is widely acknowledged; they highlight areas where knowledge is lacking, provide estimates of the relative influence of different factors within the system, and form a cohesive framework for characterisation, analysis and visualisation of results. For a model or simulation to be useful in this way, it must be able to be quickly and easily adapted to the evolving problem to which it is being applied. A generic approach to plant root architecture and microbial ecology description would clearly be a considerable advantage in achieving such flexibility. This development is discussed in the next chapter.

8.3 Parameterisation and Validation

Parameterisation of a model with such a large number of variables, many of which are difficult to measure, is a major exercise. The variables required are of course well defined, although before any major field work was undertaken, the current "ballpark" figures should be used for sensitivity analysis to determine where the most intensive effort is required.

It is probable that much of the required information has already been collected as part of other research. For example, the armillaria distribution maps (Figure 7.1, [66]), contain information about branching angles (azimuths), branching frequencies and positions that could be extracted. Considerable effort should be invested in sourcing similar studies (some of which have already been located in various personal communications). The resources required for even small scale excavation are considerable, so the time required to convert output from other studies to a suitable form will almost certainly be justified.

8.4 Visualisation

The importance of interactive, three-dimensional visualisation software became apparent early in the simulation's development. A rapid and effective technique for verifying the simulation's behaviour has an enormous impact on the speed and efficiency of model development. Non-graphical alternatives exist, but are generally extremely cumbersome. For example, routines to report total root lengths within a defined rectangular space could be applied to the generated root system, but determining the expected values for each verification step would be very time consuming. And such routines would themselves require thorough verification.

With colour and simple animation, three-dimensional line drawings (Figure 8.1) give a concise view of root system shape, root growth, fungal distribution, fungal growth and fungal interaction. Visualisation routines are also an important part of the completed simulation, both for the preparation of input, and in the interpretation of output. The generated root system can be viewed from any angle, although views from directions other than normal plan and elevation are often difficult to interpret (Figure 8.1). Raytracing has been used to produce more realistic visualisations (Figures 3.9 and 5.7).

Figure 8.1: A computer generated root architecture, plan and elevation.

8.5 General notes

8.5.1 Lists vs. Recursion

When processing a branching structure like a root system, there are two possible methods for dealing with the higher order branches encountered while processing of the "current" branch is incomplete. In the recursive case, processing of the current branch is postponed, and the higher order branch is processed. The processing of the higher order branch may in turn be postponed by a yet higher order branch (Figure 8.2 (a)). Strictly, recursive processes use a list, essentially a LIFO (Last In First Out) list whose entries are references to "the root that was being processed before the processing of the current root began". However, this list is created transparently by the programming language, and specific list handling code need not be written. Probably for this reason, recursive algorithms are often favoured by programmers.

The alternative is to build a list of unprocessed higher order roots as they are encountered, and then to process them only after the current root has been completely processed. The list may be either a LIFO or FIFO (First In First Out) type (Figure 8.2 (b) and (c)). The distinction between the list based or the recursive approach, in terms of root system simulation, is in the state of "other" roots relative to the "current" root. In the recursive case, other roots may be either completely defined, partly defined, or as yet undefined. Whereas in the list based case, other roots may only be completely defined or as yet undefined.

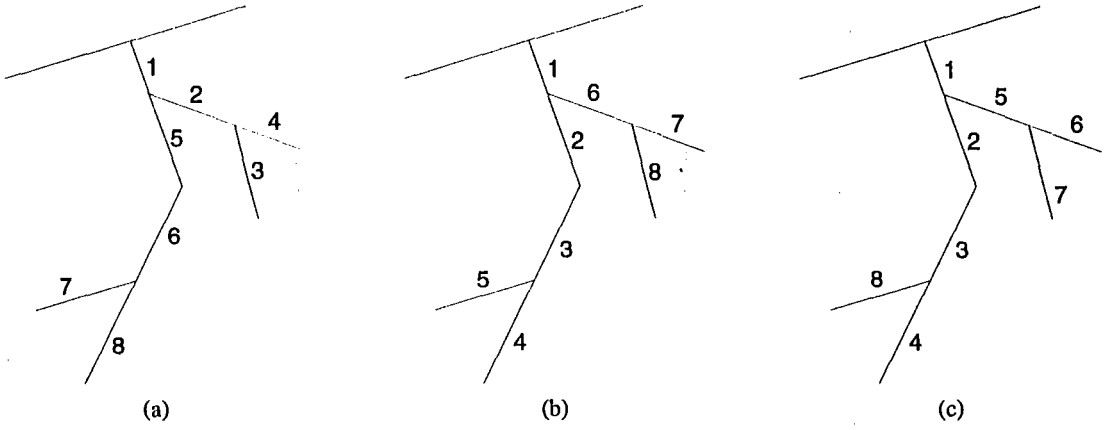


Figure 8.2: Internode processing order for (a) recursive, (b) LIFO list and (c) FIFO list algorithms.

The possibility of the existence of partly defined roots can become a problem when some form of relationship between roots is of interest. In the list based case the complete set of inter-root relationships can be examined as the roots are processed, as each root added can safely "relate" with all previous roots (none of which will be only partly defined). In the recursive case partly processed roots may prevent inter-root relationships being fully characterised without a second round of processing. The system developed here currently uses the recursive approach, and it seems unlikely that the minor increase in program run-time that may arise from occasional requirement of two or more processing sequences justifies the introduction of the added layer of complexity involved in the list based case. It is worth noting that an object oriented implementation (section 9.3.1) may well allow the two techniques to be used interchangeably, or at least allow the list based approach to be used transparently.

8.6 Summary

Initial research and implementation seems to indicate that a mechanistic, spatial simulation of the type developed will be a useful and adaptable tool for assessing plant-root pathosystem management options. The three dimensional structural envelope and node based description of root system architecture and growth developed here appears to be both viable and practical. The long timescales and largely inaccessible systems involved make computer simulation a valuable tool not only for predicting the effects of proposed schemes, but for targeting research areas of particular importance, so that research resources may be used as effectively as possible. A generic, probably object oriented, implementation of the simulation technique is considered to be the most appropriate means to such an end.

Chapter 9

Future Directions

9.1 A generic tree description

Some simulations of plant architecture have been extremely abstracted, so that the plant features only as a list of sub-units [4]. In its current form, the root system architecture description scheme developed here (sections 3.4.1 and 5.4) is reasonably flexible. However, altering anything other than simple numeric parameters requires basic to moderate (depending on the complexity of the alteration) knowledge of the C programming language.

An early goal in the continued development of the system would be the specification of a generic root architecture description syntax that allowed easy modification by users with only a limited background in computing, but still permitted the representation of complex root system features. An entry format that allows the root system to be described in natural terms would probably be most successful, although some mathematical abstractions are unavoidable. The simulation's current reliance on the use of a fully fledged programming language to describe some of the more intricate elements of root system structure is perhaps an indication of the scope of the problem. However, the real world view encouraged by object-oriented programming may result in a root architecture model more amenable to user friendly input specification.

The level of flexibility envisaged is quite high, for instance it should be possible to develop a scripting language that supports the creation of arbitrary ecosystem parameters. Take the hypothetical example of a low phosphorus soil in which plant disease resistance is improved by increased phosphorus levels, which can be conferred by a mycorrhizal fungus. Even without

any provision for this situation in the simulation code proper, a fully fledged generic simulation framework could allow the definition of an abstract "resource" or "substance" variable. The value could then be modified by mycorrhizal biomass, which would be modelled by the simulation, allowing the value to be included in the rule that was defined to determine the outcome of plant / pathogen interactions. For purely illustrative purposes, the following speculative configuration script fragment outlines the above procedure.

```
substance plant.phos = 1.
```

Define a new variable, called "plant.phos".

```
plant.phos = plant.phos + biomass(myc) / 0.25
```

Modify with mycorrhizal biomass.

```
plant.resist = plant.resist + 0.15 * plant.phos
```

Use the variable `plant.phos` to modify another variable, `plant.resist`, which is compared with another variable, say `phytophthora.virulence`, to determine if a plant / pathogen encounter (predicted by the root architecture aspect of the simulation) results in infection.

Strictly speaking the `plant.phos` variable could be eliminated from the example above, as it's a simple function of `biomass(myc)`. However, given the above framework, the addition of an "irregular event" list would be trivial. If this list contained an event that incremented `plant.phos` to simulate the application of phosphorus fertiliser, `plant.phos` would no longer be a purely derived function. A fertiliser addition or other perturbation from the steady state would also require the addition of a decay term.

9.2 Improved physiological submodels

Currently the impact of root physiology on root system morphology is modelled implicitly by the assumption that the root morphologies used to parameterise the simulation incorporated all relevant physiological effects. The same assumption is applied to the physical effects of different soil horizons. It is possible that the re-development of the root architecture specification submodel will result in a system more flexible at the individual root level. The aim would be to be able to model root development characteristics influenced by unexpected events occurring during the simulation. For example mildly diseased roots may be "stunted", while healthy roots

of trees which have lost a significant proportion of their root system to disease may exhibit increased branching frequency. Such enhancements would require more detail in the "status record" discussed in section 3.3.1, and more intelligence in the algorithms that manipulate that record. These changes are not seen as replacing the current purely structural approach, but as complimenting it.

Another area in which physiological aspects could perhaps be addressed more thoroughly is the resolution of plant–fungi and fungi–fungi interactions. The present "infection score" based approach (section 3.9.1) is certainly workable, and in the *Pinus radiata* / armillaria test case a more complex system may not be justified by the available information. However, in a more generic system a simple rule based approach may be more appropriate. The root status description syntax that would be required would hopefully be sufficiently similar to that used for root system architecture specification to allow both to be developed in parallel. While there is no specific reason that these features cannot be added in a non-object oriented implementation, the shift in emphasis towards the "problem domain" (physical system) afforded by object-oriented environments would probably be a significant advantage.

9.3 Software development

9.3.1 Programming language

Experience gained in the production of this simulation, and in work on related projects, suggests that the advantages conferred by the object oriented features of the C++ programming language would justify the time required to re-code the simulation in C++. As the overall framework of the simulation appears to be sound and most of the required mathematical algorithms have already been developed, the re-coding could probably be completed in four to five months. The simulation represents the root system as a linked list of nodes, all of which share positioning, intersection, and elongation behaviour. In addition, different types of nodes have specific behaviours; for example, rhizomorph initiation nodes communicate between the root system a rhizomorph structure, while trichoderma nodes modify the local susceptibility of the root system to infection by armillaria.

The object oriented paradigm supported by C++ greatly simplifies the manipulation of lists of

similar but non-identical objects. Polymorphism, the ability of objects to behave in a way appropriate to their specific class when issued instructions defined in the general classes from which they descend, conveniently isolates the code and resources required for variant behaviours, and so makes the task of extending the simulation simpler and less time consuming.

A specific example is the recursive parser routine, `parseRoot()`. The parser would be more easily maintained in an object-oriented environment; currently it has to traverse the root system structure in a large variety of subtly different ways, as it is used by almost all the root system analysis procedures in the simulation (section 3.3.4). This variation is achieved by an unwieldy number of flags and conditional clauses. Using C++, a parser class capable of basic root list navigation could be used as a parent class for a family of parsers with behaviour more precisely tailored to the analysis procedure they serve. In fact the programming related issues of list management could probably be isolated from the more central issues of root system simulation by "iterator" classes, common C++ constructs that simplify manipulation of complex data structures.

9.3.2 User interface

The addition of a Graphics User Interface (GUI) to the simulation would be reasonably straight forward, as the inputs required to access and control the simulation are already clearly defined. However, the time required to replace the existing "command line" based interface would be more readily justified in the context of a commercial development of the software, as it would add very little to the power of the simulation as an experimental tool. It is also worth noting that object oriented languages are commonly accepted as being ideal for manipulating modern GUI's, so the addition of a GUI would follow more readily after the simulation was re-coded in C++. A strong argument can be made for retaining the command line interface when the ease of re-compiling the simulation on different platforms is considered, although this does not preclude the implementation of a GUI on a platform by platform basis. C++ class libraries that allow portable GUI's to be written are available, but these are generally commercial products that could only be considered if the simulation was being developed for commercial purposes.

9.4 Possible applications

9.4.1 Other disease systems

The transmission of many soil borne plant diseases is characterised by the necessity for chance contact between roots and infectious material (as opposed to the almost inevitable contact between foliage and spores in above ground pathosystems). So a system that directly addresses root positioning, root contact, and the spatial distribution of microbial populations is clearly a sound basis for modelling a range of root disease pathosystems. Additionally, the emphasis placed on spatial aspects allows different soil water regimes and soil types at different depths, different initial inocula distributions (produced by different cultivation techniques), and alternative biocontrol application strategies to be considered. Gilligan [25] has reviewed spatial aspects of root disease systems.

It is generally accepted that there is a continuum from plant root diseases through harmless rhizosphere fungi to beneficial mycorrhizal symbionts. No major restructuring of the model would be required to investigate plant / mycorrhiza relationships or competition between fungal species with only indirect impacts on plant growth.

9.4.2 Root competition

The model structure also offers a particularly direct view of both inter- and intra-species root competition. The agro-forestry case, in which the evenly distributed root systems of grasses compete with the localised but more extensive root systems of trees, can easily be represented. The simulation naturally incorporates the shape of the tree root system, which will usually consist of a shallow horizontal component (occupying the same space as the grasses) and a deeper, vertical component (giving the tree greater access to ground water). Competition could be accurately quantified by assigning a radius of effect to each root type for each species, and dividing the soil volume into unutilised, uncontested and contested categories. Similarly, with the inclusion of spatial distribution information for nutrients and other resources would allow the efficiency with which a root system exploits a given soil volume to be indexed.

9.4.3 Root sampling strategies

Core sampling to measure root system characteristics is a notoriously indirect technique. Hughes and Gandar [32, 22] have considered the problem, particularly the deceptively large number of "zero" samples that can occur if root distribution is non-uniform. The method developed here allows the simulation of any number of different sampling strategies and the comparison of the results of each strategy with accurate whole root system statistics, all without significant time or resource requirement.

Similar information may have applications in situations where excavation, tillage or other modifications to the soil are required between sowing and harvest. For example, optimum placement of slow release fertilizers, or minimum safe distance for additional planting without damaging existing roots could be derived given sufficient confidence in the root system architecture characterisation.

9.4.4 Visualisation

As discussed in section 8.4, the development of the simulation described here requires sophisticated computer visualisation techniques, if time consuming indirect verification is to be avoided. Having developed these techniques, the software becomes a useful tool for producing images of systems that are seldom seen in their entirety. During development of the system, workers with extensive backgrounds in soil borne diseases, and plant root structural and nutrient uptake roles, often commented that root system form was not something they had considered in detail. The opportunity to view and manipulate root structures may be helpful in many situations where concepts of root system dynamics are being developed. It follows that the system could also be applied in educational contexts, allowing students to alter system parameters and observe the effects on root structure and microbial population distribution.

Publications

The following publications have arisen from this work:

T.N. Brown and D. Kulasiri
Simulation of *Pinus radiata* root system structure for ecosystem management applications.
Simulation 1994 62(5):286–294.

T.N. Brown and D. Kulasiri
Validating models of complex, stochastic, biological systems.
Ecological Modelling 1995 in press.

T.N. Brown
Programming Plants
New Zealand Science Monthly 1993 4(6):11.

T.N. Brown
Mapping and Modelling 3-dimensional Tree Root Architecture.
Conference on Engineering in Agriculture, Lincoln University, 1994 Paper No. SEAg 94/021.

Acknowledgments

I would like to thank my Supervisor, Dr. Don Kulasiri, for his excellent supervision, support and facilitation, my Associate Supervisor, Dr. Roy Gaunt, for his assistance and guidance, and my Technical Advisor, Dr. Robert Hill, for his input on biological topics. Thanks are also owed to Dr. Alex Watson, who supplied the root maps used as a basis for some aspects of the simulation, and Dr. Fred Baker and Stefan Zeglen, who supplied the armillaria maps in chapter 7. Dr. Alan McKinnon and the Centre for Computing and Biometrics, Lincoln University, provided valuable support and employment. Dr. Keith Pollock, Sue Thompson and others in the Field Services Centre, Lincoln University, greatly assisted with the field work undertaken.

Bibliography

[1] Anon. *New Zealand Forestry Bulletin*, page 5, December 1991.

[2] Anon. *Forestry Facts and Figures 1993*. New Zealand Forest Owners Assoc. Inc., New Zealand, 1993.

[3] J.M. Balneaves and P.J. De La Mare. Root patterns of *Pinus radiata* on five ripping treatments in a Canterbury forest. *New Zealand Journal of Forestry Science*, 19(1):29–40, 1989.

[4] A.D. Bell, D. Roberts, and A. Smith. Branching patterns: the simulation of plant architecture. *Journal of Theoretical Biology*, 81:351–375, 1979.

[5] W.J. Bloomberg. A model of damping off and root rot of Douglas fir seedlings caused by *Fusarium oxysporum*. *Phytopathology*, 69(1):74–81, 1979.

[6] W.J. Bloomberg. Model simulations of infection of Douglas fir seedlings by *Fusarium oxysporum*. *Phytopathology*, 69(10):1072–1077, 1979.

[7] T.N. Brown and D. Kulasiri. Simulation of *Pinus radiata* root system structure for ecosystem management applications. *Simulation*, 62(5):286–294, 1994.

[8] T.N. Brown and D. Kulasiri. Validating models of complex, stochastic, biological systems. *Ecological Modelling (in press)*, 1995.

[9] W.C. Carlson, C.A. Harrington, P. Farnum, and S.W. Halgren. Effects of root severing treatments on loblolly pine. *Canadian Journal of Forestry Science*, 18:1376–1385, 1988.

[10] M.A. Castellano and J.M. Trappe. Ectomycorrhizal formation and plantation performance of Douglas fir nursery stock inoculated with rhizopogon spores. *Canadian Journal of Forest Research*, 15(4):613–617, 1985.

[11] C.G. Shaw et al. Cultural characteristics and pathogenicity to *Pinus radiata* of *Armillaria novae-zelandiae* and *A. limonea*. *New Zealand Journal of Forestry Science*, 11(1):65–70, 1981.

[12] D.A. Charles-Edwards, D. Doley, and G.M. Rimmington. *Modelling plant growth and development*. Academic Press, Australia, 1986.

[13] M. Chu-Chou and L.J. Grace. Cultural characteristics of *Rhizopogon spp.* associated with *Pinus radiata* seedlings. *New Zealand Journal of Botany*, 22:35–41, 1984.

[14] M. Chu-Chou and L.J. Grace. Comparative efficiency of the mycorrhizal fungi *Laccaria laccata*, *Hebeloma crustuliniforme* and rhizopogon species on growth of radiata pine seedlings. *New Zealand Journal of Botany*, 23:417–424, 1985.

[15] M. Chu-Chou and L.J. Grace. Mycorrhizal fungi of *Pinus radiata* planted on farmland in New Zealand. *New Zealand Journal of Forestry Science*, 17(1):76–82, 1987.

[16] M. Chu-Chou and L.J. Grace. Mycorrhizal fungi of radiata pine in different forests of the North and South Islands in New Zealand. *Soil Science and Biochemistry*, 20(6):883–886, 1988.

[17] H.G. Cutler and R.A. Hill. Natural fungicides and their delivery systems as alternatives to synthetics. In C.L. Wilson and M.E. Wisniewski, editors, *Biological Control of Postharvest Diseases. Theory and Practice*, chapter 12, pages 135–152. CRC Press Inc., London, 1994.

[18] P.R. Darrah. Models of the rhizosphere. II. a quasi three-dimensional simulation of the microbial population dynamics around a growing root releasing soluable exudates. *Plant and Soil*, 138(2), 1991.

[19] A.J. Diggle. ROOTMAP – a model in three-dimensional coordinates of the growth and structure of fibrous root systems. *Plant and Soil*, 105:169–178, 1988.

[20] N.I. Federov and I.N. Bobko. Protection of pine stands from root rot caused by honey fungus. I. Chemical control. *Mikologiya i Fitopatologiya*, 22(3):255–261, 1988.

[21] N.I. Federov and I.N. Bobko. Protection of pine stands from root rot caused by honey fungus. II. Biological measures. *Mikologiya i Fitopatologiya*, 22(5):456–460, 1988.

[22] P.W. Gandar and K.A. Hughes. Kiwifruit rooting sustems: 1. Root length densities. *New Zealand Journal of Experimental Agriculture*, 16:35–46, 1988.

[23] J. Garbaye and G.D. Bowen. Effect of different microflora on the success of ectomycorrhizal inoculation of *Pinus radiata*. *Canadian Journal of Forest Research*, 17(8):941–943, 1987.

[24] S.D. Garret. Rhizomorph behavior in *Armillaria mellea* (vahl) quél. 2. logisitics of infection. *Annals of Botany*, 20(78):193–209, 1956.

[25] C.A. Gilligan. Modelling of soil bourne pathogens. *Annual Review of phytopathology*, 21:431–460, 1983.

[26] R. Hartig. Important diseases of forest trees. *Phytopathological Classics*, 12:12–41, 120pp., 1974.

[27] R. Henderson, E.D. Ford, and E. Renshaw. Morphology of the structural root system of Sitka spruce: 2. Computer simulation of rooting patterns. *Forestry*, 56(2):137–153, 1983.

[28] R. Henderson, E.D. Ford, E. Renshaw, and J.D. Deans. Morphology of the structural root system of Sitka spruce: 1. Analysis and quantitative description. *Forestry*, 56(2):121–135, 1983.

[29] Robert Hill, 1992. Personal communication, see [17].

[30] I.A. Hood. Armillaria root disease in New Zealand forests. *New Zealand Journal of Forestry Science*, 19(2):180–197, 1989.

[31] I.A. Hood and J.C. Sandberg. Occurrence of armillaria rhizomorph populations in the soil beneath indigenous forests in the Bay of Plenty, New Zealand. *New Zealand Journal of Forestry Science*, 17(1):83–89, 1987.

[32] K.A. Hughes and P.W. Gandar. Kiwi friut rooting systems: 2. Root weights. *New Zealand Journal of Crop and Horticultural Science*, 17:137–144, 1989.

[33] D.S. Jackson and J. Chittenden. Estimation of dry matter in *Pinus radiata* root systems. 1. Individual trees. *New Zealand Journal of Forestry Science*, 11(2):164–182, 1981.

[34] R.L. James, C.A. Stewart, and R.E. Williams. Estimating root disease losses in northern Rocky Mountain national forests. *Canadian Journal of Forest Research*, 14(5):652–655, 1984.

[35] J.A. Klitscher. Mortality trends in *Pinus radiata* in the Rotorua region. *New Zealand Forestry*, pages 23–25, February 1987.

[36] L.C. Kuiper and M.P. Coutss. Spatial disposition and extension of the structural root system of Douglas Fir. *Forest Ecology and Management*, 47:111–125, 1992.

[37] W.A. Kurz and J.P. Kimmins. Analysis of some sources of error in methods used to determine fine root production in forest ecosystems: A simulation approach. *Canadian Journal of Forest Research*, 17(8):909–912, 1987.

[38] W.J. Libby. "Connectedness". A talk delivered to the New Zealand Institute of Forestry, Canterbury section, and the Royal Society of New Zealand, Canterbury Branch, on April 21, at the University of Canterbury, New Zealand, by Professor William J. Libby, University of California, Berkeley, April 1993.

[39] D.R. Lungley. The growth of root systems, a numerical computer simulation model. *Plant and Soil*, 38:145–159, 1973.

[40] H.J. Lutz, J.B. Ely, and S. Little. The influence of soil profile horizons on root distribution of white pine (*Pinus strobus* l.). Bulletin 44, Yale University School of Forestry, New Haven, 1937.

[41] M. MacKenzie. Infection changes and volume loss in a 19 year old *Pinus radiata* stand affected by armillaria root rot. *New Zealand Journal of Forestry Science*, 17(1):100–108, 1987.

[42] M. MacKenzie and C.G. Shaw III. Spatial relationships between armillaria root rot of *Pinus radiata* seedlings and the stumps of indigenous trees. *New Zealand Journal of Forestry Science*, 7(3):374–383, 1977.

[43] C. O'Loughlin and A. Watson. Root-wood strength deterioration in radiata pine after clear felling. *New Zealand Journal of Forestry Science*, 9(3):284–293, 1979.

[44] R.F. Patton and A.J. Riker. Artificial inoculation of pine and spruce trees with *Armillaria mellea*. *Phytopathology*, 49:615–622, 1959.

[45] K.M. Reynolds. Simulation and spread of *Phytopthora cinnamomi* causing a root rot of Fraser fir in nursery beds. *Phytopathology*, 76(11):1190–1201, 1986.

[46] N.F. Robertson. Studies on the mycorrhiza of *Pinus sylvestris*. I. The pattern of development of mycorrhizal roots and its significance for experimental studies. *New Phytologist*, 53:253–282, 1954.

[47] D.A. Rose. The description of the growth of root systems. *Plant and Soil*, 75:405–415, 1983.

[48] L.F. Roth, C.G. Shaw III, M. MacKenzie, and F. Crockett. Early patterns of armillaria root rot in New Zealand pine plantations converted from indigenous forest - an alternative interpretation. *New Zealand Journal of Forestry Science*, 9(3):316–323, 1979.

[49] D. Sanantonio and E. Sanantonio. Effect of thinning on production and mortality of fine roots of *Pinus radiata* on a fertile site in New Zealand. *Canadian Journal of Forest Research*, 17(8):919–928, 1987.

[50] D. Sanantonio and E. Sanantonio. Seasonal changes in live and dead fine roots during two successive years in a thinned plantation of *Pinus radiata* in New Zealand. *New Zealand Journal of Forestry Science*, 17(2):315–328, 1987.

[51] T.M. Schlicter, R.R. Van der Ploeg, and B. Ulrich. A simulation model of the water uptake of a beech forest: testing variations in root biomass and distribution. *Zeitschrift fur Pflanzenernahrung und Bodenkunde*, 146(6):725–735, 1983.

[52] C.G. Shaw III and S. Calderon. Impact of armillaria root rot in plantation of *Pinus radiata* established on sites converted from indigenous forest. *New Zealand Journal of Forestry Science*, 7(3):359–373, 1977.

[53] A. Somerville. Root anchorage and morphology of *Pinus radiata* on a range of ripping treatments. *New Zealand Journal of Forestry Science*, 9(3):294–315, 1979.

[54] G.R. Stanosz and R.F. Patton. Armillaria root rot in Wisconsin aspen sucker stands. *Canadian Journal of Forest Research*, 17(9):995–1000, 1987.

[55] J.R. Stephenson. *People and Pines; Industrial Forestry in New Zealand.* Ecumenical Secretariat on Development., Auckland, New Zealand., 1981.

[56] D.J. Stewart, C.J. Amlaner, and C.C. Barnett. Harvest: A generalised animal population growth simulation. *Simulation*, 59(1):57–64, 1992.

[57] A.R.D. Trewin. A fully integrated system for planting bare-root seedlings of Radiata Pine in New Zealand. *Quality Assurance New Zealand*, 16:7–11, 1989.

[58] A.R.D. Trewin and P.M. Kirk. Planting bare rooted seedlings of radiata pine on difficult terrain. In *Harvesting and Re-establishment of Difficult Terrain*, Hastings/Gisbourne, June 1992. Logging industry research seminar.

[59] A.R.D. Trewin and A.A. Twaddle. Challenges to the re-establishment of radiata pine plantations. Number 15 in Council of Forest Engineering Meeting, Little Rock, Arkansas, August 1992. Council of Forest Engineering.

[60] J.B. van der Pas. A statistical appraisal of armillaria root rot in New Zealand plantations of *Pinus radiata*. *New Zealand Journal of Forestry Science*, 11(1):23–36, 1981.

[61] J.B. van der Pas, I.A. Hood, and M. McKenzie. Armillaria root rot. Forest pathology in New Zealand leaflet series 4, 1983.

[62] A. Watson and C. O'Loughlin. Structural root morphology and biomass of three age-classes of *Pinus radiata*. *New Zealand Journal of Forestry Science*, 20(1):97–110, 1990.

[63] B. Wickham and M. Watson. New Zealand: a good option for wood fibre processing. *New Zealand Forestry Bulletin October*, pages 4–5, October 1991.

[64] K.K.Y. Wong et al. Difference in the colonisation of *Pinus banksiana* roots by sib-monokaryotic and dikaryotic strains of ectomycorrhizal *Laccaria bicolor*. *Canadian Journal of Botany*, 67(6):1717–1726, 1989.

[65] T.H. Wu, D.P. Bettadapura, and E.P. Beal. A statistical model of root geometry. *Forest Science*, 34(4):980–997, 1988.

[66] S. Zeglen. An investigation of armillaria root disease in a lodgepole pine stand. Master's thesis, Utah State University, 1991.

Appendix A

File formats

A.1 Configuration file `config.rs`

This file can be read and edited by anyone familiar with the use of a text editor. Each entry is comprised of one or more lines of text describing the value represented by the entry, followed by a unique id-tag and the value referred to on a separate line. For example, the entry:

```
Length for 0th order root
P25Y0RoLe: 0.100000
```

would indicate that the length of the 0th order root is 0.1 m. This can be changed by editing the numeric value after the id-tag `P25Y0RoLe:`, but the tag, and it's position at the beginning of the line, can not be altered. The whole entry may be deleted, in which case the simulation software will request a new value for the parameter next time it is run. The entry for this new value will be appended to the end of the configuration file.

Some of the distributions that use the configuration file may place several entries in it, for example:

```
Enter number of entries for rectangular freq. dist'n
for 1st order init. angle (branch angle from order 0)
P25Y0B_An_n: 6

Lower bound of frequency interval 0
```

```
P25Y0B_An_x00: -1.570796

Weighting of frequency interval 0
P25Y0B_An_w00: 30.000000

Lower bound of frequency interval 1
P25Y0B_An_x01: -1.413720

Weighting of frequency interval 1
P25Y0B_An_w01: 16.000000

[etc.]
```

These entries should either be deleted together, or edited in position, as deleting only one causes the entries to become separated in the configuration file and may cause the simulation to request values whose description is ambiguous when out of context.

A.2 Filter files

Filter files store the maximum and minimum values for the root node filters described in 3.3.2. Two numbers (minimum and maximum) are read from the beginning of each line, the remainder of the line is ignored, and may be used for commenting on the meaning of the numbers.

The order of the entries is as follows:

Default minimum value	Default maximum value	Element of `StatusType` structure	Meaning
0	6	n	root order
0	50	phase	root's physiological growth phase
0	100	pLen	position of root's origin on parent root
0	100	fLen	root's final length when mature
-10000	10000	pnt.x	x position of node
-10000	10000	pnt.y	y position of node
-10000	10000	pnt.z	z position of node
-100	100	an	angle of internode
-100	100	az	azimuth of internode
0	100	tot	root's total of nodes of this type
0	100	cur	nodes of this type so far encountered
-100	100	fLevel[Armillaria]	level of fungal infection at this node
-100	100	fLevel[Rhizopogon]	
-100	100	fLevel[Trichoderma]	
-100	100	cLevel[Armillaria]	amount of fungal material in this lesion
-100	100	cLevel[Rhizopogon]	
-100	100	cLevel[Trichoderma]	
-100	100	tLevel[Armillaria]	amount of fungal material in this lesion and
-100	100	tLevel[Rhizopogon]	proceeding lesions
-100	100	tLevel[Trichoderma]	
-10000	10000	fMod[Armillaria]	environmental modification factor for each fun-
-10000	10000	fMod[Rhizopogon]	gus on this root
-10000	10000	fMod[Trichoderma]	

The simulation expects two of these filter files to be available when it is run. `def_filt.rs` is the default filter, which should generally consist of extreme minimum and maximum values which are unlikely to filter out any node. `rhz_filt.rs` identifies internodes infected with a level of the fungus armillaria sufficient to produce rhizomorphs.

A.3 Ground height file `ground.rs`

`xdiv ydiv`	The number of divisions of the grid along the x and y axis
`xorg yorg`	The x, y origin of the grid
`xsz ysz`	The x, y extent of the grid
`zval zval zval ...`	`xdiv`×`ydiv` z-values, starting at the top left hand corner of the grid (`xorg,yorg+ysz`)

The program function `ground(x,y)` reports the z-value for an x,y pair. The function loads the file `ground.rs` the first time it is called.

A.4 Display colour file `rgb.rs`

On the Amiga™ platform this file lists RGB triplets with values in the range 0-15, to define the colours used in interactive graphics displays. On the IBM-AT™ platform the behaviour is the same, unless a ega-mono graphics adaptor is detected, in which case `rgb.rs` behaves as a lookup table to control the order in which default colours are used. For example, given that the first five entries were 0,1,2,1,4, colour 1 would be displayed when the colour specified was 1 or 3.

Appendix B

Program notes and instructions

A 3.5 inch disk is supplied with this thesis, containing source code, configuration files, some demonstration images, and a runnable version of the program for IBM™ PC compatible computers. The disk is formatted as 1.44 Mbyte MSDOS™ media, which can be read by Macintosh™ and Unix platforms with appropriate hardware / software.

B.1 Compiling the program

It is the intention of the author to re-implement the program in a more generic form, in the C++ programming language. If you want to use this software for anything other than a cursory evaluation, please contact the author for details of software's current state. Either email, to:

`brownt1@lincoln.ac.nz`

or

`tb@tbnode.equinox.gen.nz`

or standard mail to:

Terry Brown
c/o Centre for Computing and Biometrics
Lincoln University
New Zealand

Only the MSDOS version, compiled under Borland™ C++, has a graphical display, the Unix version will however save PostScript files which can be viewed on-screen while the simulation is running in many Unix environments. The code has also been compiled, with a graphical display, on the Amiga™ platform with both GCC and Manx Aztec™ C compilers. Note that the file `rs_numb.c` is not valid C code, but input for the program `numbmake` (also included), which converts it to `rs_auto.c`, if the supplied makefile is used. The program should compile readily with GCC 2.5.8 or higher, requirements of other compilers may vary. In particular the function `dumpInput()`, in `rs_io.c`, may need altering, this function flushes `stdin` without pausing until return had been pressed. Compiling the graphical version with Borland C++ 4.0 is complicated by a minor bug in the Borland IDE (Integrated Development Environment).

B.2 Running the program

If the program is compiled under Unix, the executable should be placed in a directory with the files found in the `run` subdirectory of the disk. Most of the following instructions should apply, apart from those involving graphical displays.

Note that every time the program is run it saves a logfile name `rs-log.xxx` where `xxx` is a number chosen to avoid overwriting any previous logfiles, as these files accumulate they may need to be periodically deleted.

The program accepts the following command line parameters:

Parameter	Effect	Default	Bytes per item
`-Rx`	Allow up to x roots	5000	68
`-Nx`	Allow up to x nodes	30000	44
`-Sx`	Allow up to x stems (trees)	10	44
`-Lx`	Allow up to x root list entries	5000	8

One root list entry is required per root passing through a voxel lattice element, an additional 500 entries should be allowed for general use by the program. The default values need only be changed if the program halts reporting one of these resources has been exhausted.

For MSDOS only – copy the files *and* the directory in the run subdirectory onto a harddisk, and

enter "go", for example:

```
mkdir c:\root-sim
c:
cd \root-sim
xcopy /s a:run .
go <return>
```

This calls a batch file that runs the program with command line parameters which limit the total number of roots and nodes to values the MSDOS memory allocation system can handle. A mouse driver should be loaded before the program is run, `MOUSE.SYS` is known to work.

B.2.1 Entering commands

The menu system is self-explanatory. After specific parameters have been entered, most commands prompt for an event time. Just pressing return causes the command to take effect immediately, entering a time (a real number, in days) adds the command to the simulation script. Multiple events scheduled for the same time will be performed in the order they were entered. Events may be scheduled for a time that has already passed, all such events will be performed as soon as simulation script processing is restarted.

Building a root system Choose option 1 from the main menu, and option 1 from the root submenu. Enter a tree number, probably 0. Enter a location, this is an x,y,z location in metres, 0,0,0 is suitable for most applications. Select a tree type, type 2 is the *Pinus radiata* parameter set used for most of this project. A "C-tree" is a coffee mug tree, which was created to test the versatility of the root morphology description system. Answer "n" to the prompt for making a grid of trees.

Viewing a root system Select option 9 from the root submenu. The view initially presented is a plan view of the root system. There is a three-dimensional cursor box (Figure B.1) associated with most graphics displays. The cursor's appearance is consistent with it being made out of wire. The presence or absence of this cursor indicates whether or not the display is in "cursor mode", various keys have different functions in and out of this mode. The keyboard controls are

listed in tables B.1 and B.2. Note that "space" is used both to exit cursor mode and to re-enter it, typically after rotating/panning/zooming the cursor in cursor mode space is pressed twice to cause the image to be re-displayed from the new point of view. The root editing features were implemented as minimalistically as possible, and their use is quite complicated and idiosyncratic, so it is not practical to provide a simple demonstration of their use.

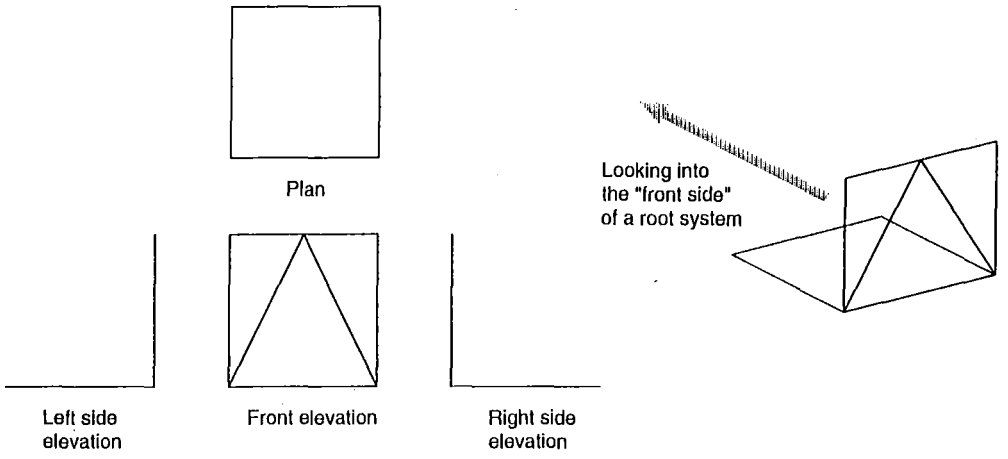


Figure B.1: Shape of three-dimensional cursor

B.2.2 Simulating cross infection – an example

To simulate infection from a dead root system to a group of four seedlings, as illustrated in Figure B.2, the following script (supplied in the file `crossdmo.rs`) could be used.

Key	Effect
e	Align cursor to front elevation view
1	Zoom in
2	Zoom out
s	rotate anti-clockwise around y axis
d	rotate clockwise around y axis
q	rotate anti-clockwise around z axis
w	rotate clockwise around z axis
a	rotate clockwise around x axis
z	rotate anti-clockwise around x axis
i	move up, +z
m	move down, -z
j	move left, -x
k	move right, +x
l	move out, -y
o	move in, +y
Space	exit cursor mode
p	save image as postscript file
H	save image as hpgl file
C	display current image size
3	similar to above
5	display axis
g	display grid
X	pick point with mouse and print coordinates
T	load PBM format backdrop image
S	toggle PBM display mode
I,J,K,M	fine alignment of PBM image as per lower case keys
:	increase perspective effect (not useful)
"	decrease perspective effect
?	zero perspective effect
,	decrease pan step size
.	increase pan step size
<	decrease rotation step size
>	increase rotation step size

Table B.1: Keyboard controls in cursor mode

Key	Effect
0	(zero) exit display mode
Space	re-draw image and re-enter cursor mode
Z	reset zoom to include whole/all root system(s)
F	load a filter file (section 3.3.2) and display only roots matching that filter
C	load a filter file (section 3.3.2) and display roots matching that filter in another colour
K	delete a root
M	move a node
B	add a node
X	trim a root
/	add a root
T	load and display ground topography
Space	redraw image

Table B.2: Keyboard controls in non-cursor mode

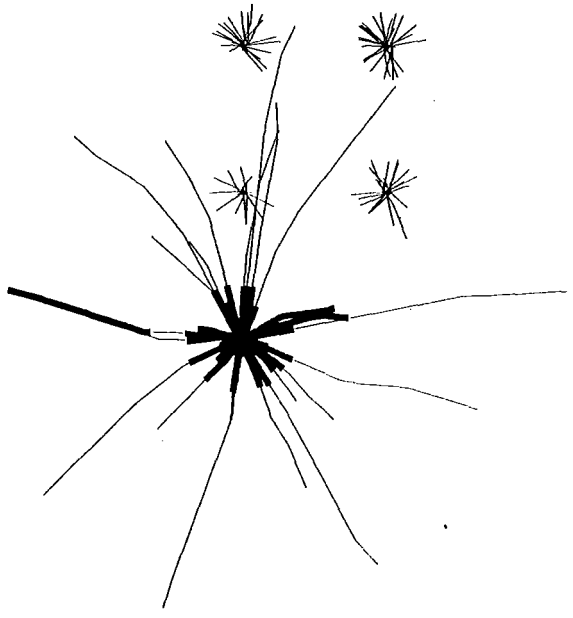


Figure B.2: Demo simulation layout

At 0 days:

- Create a grid of type 2 trees 2 by 2 with 1.5 m spacing from 0,0,0 (trees 0 to 3)
- Create a type 2 tree at 0,-1.5,0 (tree 4)
- Remove fungi from trees 0 to 3
- Age tree 4 to the Recently Dead stage and adjust its roots to their mature lengths
- Build a 10 by 10 by 10 voxel lattice to index the root system
- Set the timestep to 10 days

At 30 days and then every 30 days for 20 repetitions:

- Check for cross-infection between root systems

At 60 days and then every 60 days for 10 repetitions:

- Sample the root systems of trees 0-3 with a 10 m deep by 100 m radius core sample.
- Save an image of the root system without panning, rotating or zooming

B.3 Viewing the demonstration images

These images are taken from a longer, animated demonstration available from the author either in `.FLI` computer animation format with MSDOS player software, or on video. `FLI` format animation players are available for most operating systems, for example `xanim` for XWindows systems. To view the images using MSDOS, enter the following commands (assuming the supplied disk is in the "A:" drive).

```
a:
cd \demo
msview
```

The directory demo contains a list of GIF™ images, ordered by their file names, so any image viewing program should be able to display them. For example, to use the popular `xv` viewer under XWindows, if the disk is mounted as "`/a`":

```
cd /a/demo
xv *.gif
```

Press space to have `xv` step from one image to the next.

Appendix C

Parameter files for *Pinus radiata*.

C.1 rs_numb.c

The file rs_numb.c is used to generate *Pinus radiata* and rhizomorph "root systems", as described in section 5.7.1. This pseudo C code is translated to valid C by the program numbmake, and included in the module rs_dists.c. Calls to readParam() read values from the file config.rs, included at the end of this appendix.

```
    #define YEAR_EFFECT 1 /* (1+sin(yrTime*2.*pi)) */
    #define HORIZ_PRIMARY (st->an > -pi/8.)
    #define FOREVER 1000000.
    #define GET(TYPE,VAR) VAR = va_arg(vp,TYPE)
    #define INIT static int init = 0; if (!init) { init = 1;
    #define ENDINIT }
    /* #define CYL_BOLE */

    static double dpth25;     /* config variables used by several   */
    static double rad25;      /* functions for the 25 year old pine */

Pine25,TreeDefault,Bend,Number:
    x = 0.0;
Pine25,TreeDefault,Bend,Angle:
    x = st->an;               /* i.e. no change */
Pine25,TreeDefault,Bend,Azimuth:
    x = st->az;
Pine25,TreeDefault,Bend,Position:
    x = 0.0;
Pine25,TreeDefault,Branch,Number:
    x = 0.0;
Pine25,TreeDefault,Branch,Angle:
    x = st->an;
Pine25,TreeDefault,Branch,Azimuth:
    x = st->az;
```

```
Pine25,TreeDefault,Branch,Position:
    x = 0.0;
Pine25,TreeDefault,Fungi,Number:
    x = 0.0;
Pine25,TreeDefault,Fungi,Length:
    x = 0.0;
Pine25,TreeDefault,Fungi,Azimuth:
    x = st->az;
Pine25,TreeDefault,Fungi,Angle:
    x = st->an;
Pine25,TreeDefault,Fungi,Elongation:
    x = 0.;

Pine25,InitVect,B_Vect,Angle:

    /* x = -pi/2; */
    x = 0.;

Pine25,InitVect,B_Vect,Azimuth:
    x = 0.0;


Pine25,0,Root,Length:

    #ifdef CYL_BOLE
    INIT
        readParam(
            "Depth for root bole cylinder",
            "P25Y0RoBoDp",
            "%lf",&dpth25
        );
        readParam(
            "Radius for root bole cylinder",
            "P25Y0RoBoRa",
            "%lf",&rad25
        );
    ENDINIT

    x = rad25 + 12. * (dpth25 + 2 * rad25 * sin(pi/12));
    #else
    x = 0.1;
    #endif

Pine25,0,Root,Elongation:
    #ifdef CYL_BOLE
    x = rad25 + 12. * (dpth25 + 2 * rad25 * sin(pi/12));
    #else
    x = 0.1;
    #endif

Pine25,0,Root,Lifespan:
    x = FOREVER;
    /* as this 'root' is largely ignored, its growth phase is also
```

```
        unimportant */

Pine25,0,Bend,Number:
    #ifdef CYL_BOLE
    x = 24.;
    #else
    x = 0;
    #endif

Pine25,0,Bend,Position:

    #ifdef CYL_BOLE
    if (st->cur == 1) {
        x = rad25;     /* the radius of the trunk/bole */
    } else {

        if (st->cur & 1) {
            x = dpth25;
        } else {
            x = 0.;
        }
        x += rad25;
        x += dpth25 * ((st->cur - 2) / 2);
        x += (2 * rad25 * sin(pi/12)) * ((st->cur) / 2);

    }

    x /= rad25 + 12. * (dpth25 + 2 * rad25 * sin(pi/12));
    #endif

Pine25,0,Bend,Angle:

    #ifdef CYL_BOLE
    if (st->cur > 1) {
        if ( ((st->cur - 1) / 2) & 1)
            x = st->an + pi/2.;
        else
            x = st->an - pi/2.;
    } else {
        x = 0.;
    }
    #endif
Pine25,0,Bend,Azimuth:

    #ifdef CYL_BOLE
    if (st->cur == 1)
        x = pi/2. + pi/12.;
    else
        x = st->az + pi/12.;
    #endif

Pine25,0,Fungi,Elongation:
    static double eln;
```

```
    int fType;
    INIT
        readParam(
            "Fungal elongation rate on root order 0",
            "PY250FuEl",
            "%lf",&eln
        );
    ENDINIT

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = timeStep * eln;
        break;
    default:
        x = 0.;
        break;
    }
    /* the fungal status of this root itself is not currently
       relevant, but this controls the spread from the base of
       one 1st order root to another */

Pine25,0,Branch,Number:
    static double min;
    static double max;
    INIT
        readParam(
            "Minimum number of first order roots (for U(min,max))",
            "P25Y0BrNu_min",
            "%lf",&min
        );
        readParam(
            "Maximum number of first order roots (for U(min,max))",
            "P25Y0BrNu_max",
            "%lf",&max
        );
    ENDINIT
    x = UD(min,max);
    /* the number of first order laterals */
Pine25,0,Branch,Position:
    x = U(0.,1.);
    /* first order roots' origins are 0-0.1m below the surface, this
       should be increased, with later roots starting lower */
Pine25,0,B_Vect,Angle:
    static int n;
    static double *dat;
    INIT
        readParam(
            "Enter number of entries for rectangular freq. dist'n\n"
            "for first order initial angle "
            "(branch angle from order 0)",
            "P25Y0B_An_n",
```

```
            "%d",&n
        );
        dat = (double*)malloc(sizeof(double)*n*2);
        if (1) {
            int i;
            char buf[2][80];
            for (i=0; i<n; i++) {

                sprintf(
                    buf[0],"Lower bound of frequency interval %d\0",i
                );
                sprintf(buf[1],"P25Y0B_An_x%02d\0",i);
                readParam(buf[0],buf[1],"%lf",dat+2*i);

                sprintf(
                    buf[0],"Weighting of frequency interval %d\0",
                i);
                sprintf(buf[1],"P25Y0B_An_w%02d\0",i);
                readParam(buf[0],buf[1],"%lf",dat+2*i+1);
            }
        }
    ENDINIT
    x = dArrDist(n,dat);
  /*    x = dDist(
            -pi/2.,8.,
            -pi/2.*.9,16.,
            -pi/2.*.8,8.,
            -pi/2.*.2,16.,
            -pi/2.*.1,30.,
            0.,0.,
            -1.,-1.
        ); */
    /* a rectangular distribution fitted by observation, separates
       1st order roots into vertical and horizontal groups */
Pine25,0,B_Vect,Azimuth:
    x = U(-pi,pi);


Pine25,1,Root,Lifespan:
   switch (st->phase) {
   case Waiting:
       x = pDist( 0., 1.,
                  yrLength / 12, 1./143.,
                  yrLength * 8,  0.,
                  -1., -1.   );
       /* 80% of first order roots start to grow in the first
          month, the rest start over the next 8 years */
       break;
   case Growing:
       x = 8. * yrLength;
       /* 8 years to mature */
       break;
   case Mature:
```

```
        x = U(2.,6.) * yrLength;
        /* 2 to 6 years mature */
        break;
    case Senescent:
        x = 1 * yrLength;
        break;
    case Dead:
        x = 3. * yrLength;
        break;
    case Decayed:
        x = 3. * yrLength;
        break;
    default:    /* Future and Ghost phases come here */
        printf("Call to lifespan for unknown phase or "
               "illogical phase %d\n",
               st->phase);
        break;
    }
Pine25,1,Root,Length:
    x = HORIZ_PRIMARY ? U(1., 3.5):
                        U(0.5,3.) * (2.+st->tree->u)/3. ;
    /* first order length dependent on vertical or horizontal habit,
       observed from Watson 8yr */
Pine25,1,Root,Elongation:
    switch (st->phase) {
    case Waiting:
        x = 0.;
        break;
    case Growing:
     x = timeStep*YEAR_EFFECT * 2.5 / 8 / yrLength;
        break;
    case Mature:
     x = timeStep*YEAR_EFFECT * 2.5 / 8 / yrLength / 2.;
        break;
    case Senescent:
        x = 0.;
        break;
    case Dead:
        x = 0.;
        break;
    case Decayed:
     x = -1 * timeStep*YEAR_EFFECT * 2.5 / 8 / yrLength;
        break;
    default:    /* Future and Ghost phases come here */
        printf("Call to elongation for unknown phase or "
               "illogical phase %d\n",
               st->phase);
        break;
    }

Pine25,1,Branch,Number:
    x = (sqrt(SQR(st->pnt.x)+SQR(st->pnt.y)) < 0.75) ? Poisson(4.) :
        HORIZ_PRIMARY ? Poisson(1.) : Poisson(2.);
```

```
   /* first order branching determined by horizontal/vertical habit,
      observed from Watson 8 yr  Also, branching more intense
      in verticals below the stem */
Pine25,1,Branch,Position:
   x = HORIZ_PRIMARY ? U(0.0, 0.75) :
                       U(0.75, 1.0);
   /* first order branching determined by horizontal/vertical habit,
      observed from Watson 8 yr */

Pine25,1,B_Vect,Angle:
   x = HORIZ_PRIMARY ? st->an + U(-pi / 2, 0.) :
                       st->an + U(-pi / 2, 0.);
   /* Watson 16yr */

Pine25,1,B_Vect,Azimuth:
   x = HORIZ_PRIMARY ? st->az + U(-pi / 4.0, pi / 4.0) :
                       U(-pi, pi);
   /* horizontals branch +/- 45 degrees, UNFOUNDED, verticals branch
      to any compass direction, no evidence otherwise */

Pine25,1,Bend,Number:
    x = UD(2.,3.);
    if (st->an < -pi/8 && st->an > -pi*3/8)
        x += 2;
    /* make these (45 degree-ish) roots more flexible, to allow
       more chance to level out (see Pine25,1,Bend,Angle below) */
Pine25,1,Bend,Position:
   x = U(0.2, 0.9);
   /* UNFOUNDED */
Pine25,1,Bend,Angle:
   do {
       x = st->an + U(-pi / 12., pi / 12.);
      }  while (x > 0.0);
   /* random, but can't veer upwards */
   if (x > -2*pi/6. && x < -pi/6.) x += pi/12.;
   /* however, sloping roots (neither vert. or horiz.) have a
      tendency to level out to the horizontal */
Pine25,1,Bend,Azimuth:
   x = st->az + U(-pi / 10.0, pi / 10.0);
   /* UNFOUNDED */

Pine25,1,Fungi,Number:

    int fType;

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = Poisson(0.2);
        /* 20% chance a 1st order root has a A. lesion */
        break;
    default:
```

```
        x = 0.;
        break;
    }
Pine25,1,Fungi,Position:

    int fType;

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = U(0.6,1.);
        break;
    default:
        x = 0.;
        break;
    }
Pine25,1,Fungi,Length:

    int fType;

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = U(0.1,0.2);     /* lesion 10-20 cm long */
        break;
    default:
        x = 0.;
        break;
    }
Pine25,1,Fungi,Elongation:
    int fType;

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = 0.002 * timeStep;     /* faster than average root */
        /* changed from 0.002 - TB */
        break;
    default:
        x = 0.;
        break;
    }



Pine25,2,Root,Length:
   x = (st->pnt.z > -1.) ? U(0.5, 1.) : U(0.1, 0.5);
   /* tend to be shorter below 1m, from Watson 8yr,
      but actual values UNFOUNDED */
Pine25,2,Root,Lifespan:
```

```
    switch (st->phase) {
    case Waiting:
        x = 0.;
    case Growing:
        x = 0.8*yrLength;
        break;
    case Mature:
        x = 0.2*yrLength;
        break;
    case Senescent:
        x = 0.1*yrLength;
        break;
    case Dead:
        x = 0.1*yrLength;
        break;
    case Decayed:
        x = 0.2*yrLength;
        break;
    default:    /* Future and Ghost phases come here */
        printf("Call to lifespan for unknown phase or "
               "illogical phase %d\n",
               st->phase);
        break;
    }
    /* re-cycle after 1 year, UNFOUNDED */
Pine25,2,Root,Elongation:
    x = st->fLen / (2./ 12.* yrLength) * timeStep * YEAR_EFFECT;
    if (st->phase==Waiting) x *= 0.;
    if (st->phase>=Dead) x *= -0.5;
    /* grow to final length in 2 months, UNFOUNDED */

Pine25,2,Bend,Number:
    x = UD(1., 3.);
    /* looks reasonable, UNFOUNDED */
Pine25,2,Bend,Position:
    x = U(0.05, 0.9);
    /* looks reasonable, UNFOUNDED */
Pine25,2,Bend,Angle:
    x = st->an + U(-pi / 8.0, pi / 8.0);
    /* looks reasonable, UNFOUNDED */
Pine25,2,Bend,Azimuth:
    x = st->az + U(-pi / 8.0, pi / 8.0);
    /* looks reasonable, UNFOUNDED */

Pine25,2,Branch,Number:
    x = UD(1., 4.);
    /* these roots are fairly small, really only avoids the bare
       appearance the system has in their absence */
Pine25,2,Branch,Position:
    x = U(0.3, 0.9);
Pine25,2,B_Vect,Angle:
    x = U(-pi / 4.0, -pi / 3.0);
Pine25,2,B_Vect,Azimuth:
```

```
   x = U(-pi, pi);

Pine25,2,Fungi,Elongation:
    int fType;

    GET(int,fType);     /* extended argument - fungi type */

    switch(fType) {
    case Armillaria:
        x = 0.005 * timeStep;     /* 5 mm a day, 1 m ca. 6 mnth */
        break;
    default:
        x = 0.;
        break;
    }


Pine25,3,Root,Length:
   x = U(0.01, 0.1);
Pine25,3,Root,Lifespan:
   switch (st->phase) {
   case Waiting:
       x = 0.;
       break;
   case Growing:
       x = 40.;
       break;
   case Mature:
       x = 20.;
       break;
   case Senescent:
       x = 5.;
       break;
   case Dead:
       x = 5.;
       break;
   case Decayed:
       x = 10.;
       break;
   default:    /* Future and Ghost phases come here */
       printf("Call to lifespan for unknown phase or "
              "illogical phase %d\n",
              st->phase);
       break;
   }
Pine25,3,Root,Elongation:
   x = 0.05 * YEAR_EFFECT;

Pine25,3,Bend,Number:
   x = 0.;
Pine25,3,Branch,Number:
   x = 0.;
   /* below this would be fine (<5mm) roots */
```

```
    /* Stub used as a starting point for digitised trees */

Stub,TreeDefault,Fungi,Number:
    x = 0.0;
Stub,TreeDefault,Fungi,Lifespan:
    x = 0.0;
Stub,TreeDefault,Branch,Number:
    x = 0.0;
Stub,TreeDefault,Bend,Number:
    x = 0.0;
Stub,InitVect,B_Vect,Angle:
    x = -pi/2.;
Stub,InitVect,B_Vect,Azimuth:
    x = 0.0;
Stub,0,Root,Length:
    x = 0.1;
Stub,0,Root,Lifespan:
    x = FOREVER;


Rhizomorph,TreeDefault,Root,Lifespan:
   x = FOREVER;
   /* UNFOUNDED */
Rhizomorph,0,Root,Lifespan:
   x = 150;
   /* UNFOUNDED */
Rhizomorph,TreeDefault,Root,Elongation:
   x = 0.010 * timeStep;
   /* 10mm a day, UNFOUNDED */
Rhizomorph,TreeDefault,Branch,Number:
   x = 0.;
Rhizomorph,TreeDefault,Branch,Angle:
   x = st->an;
   /* doesn't change direction at branch points */
Rhizomorph,TreeDefault,Branch,Azimuth:
   x = st->az;
   /* doesn't change direction at branch points */
Rhizomorph,TreeDefault,Bend,Number:
   x = 0.;
Rhizomorph,TreeDefault,Fungi,Number:
   x = 0.;
Rhizomorph,TreeDefault,Fungi,Elongation:
   x = 0.;
Rhizomorph,TreeDefault,Fungi,Angle:
   x = st->an;
Rhizomorph,TreeDefault,Fungi,Azimuth:
   x = st->az;

Rhizomorph,InitVect,B_Vect,Angle:
   x = U(-pi/8., pi/8.);
   /* starts roughly horizontal */
Rhizomorph,InitVect,B_Vect,Azimuth:
```

```
   x = U(-pi, pi);
   /* starts in any direction */

Rhizomorph,0,Root,Length:
   x = U(2.,6.);
   /* an initial guess, to be looked up, UNFOUNDED */

Rhizomorph,0,Branch,Number:
   x = UD(0.,4.);
   /* UNFOUNDED */
Rhizomorph,0,Branch,Position:
   x = U(0.1,0.9);
   /* UNFOUNDED */
Rhizomorph,0,B_Vect,Angle:
   x = U(-pi/8., pi/8.);
   /* continues roughly horizontal */
Rhizomorph,0,B_Vect,Azimuth:
   x = st->az+U(-pi/2., pi/2.);
   /* may turn up to 90 degrees from previous course, UNFOUNDED */

Rhizomorph,0,Fungi,Number:
   x = 1.;
Rhizomorph,0,Fungi,Position:
   x = 0.;
Rhizomorph,0,Fungi,Length:
   x = -1.;

Rhizomorph,1,Root,Length:
   x = U(0.5,2.);
   /* UNFOUNDED */
Rhizomorph,1,Fungi,Length:
   x = -1.;
```

C.2 config.rs

The file config.rs is used to store frequently altered numeric values.

```
Length for 0th order root
P25Y0RoLe: 0.100000

Minimum number of first order roots (for U(min,max))
P25Y0BrNu_min: 20.000000

Maximum number of first order roots (for U(min,max))
P25Y0BrNu_max: 40.000000

Enter number of entries for rectangular freq. dist'n
for first order initial angle (branch from order 0)
P25Y0B_An_n: 6

Lower bound of frequency interval 0
P25Y0B_An_x00: -1.570796

Weighting of frequency interval 0
P25Y0B_An_w00: 30.000000

Lower bound of frequency interval 1
P25Y0B_An_x01: -1.413720

Weighting of frequency interval 1
P25Y0B_An_w01: 16.000000

Lower bound of frequency interval 2
P25Y0B_An_x02: -1.256640

Lower bound of frequency interval 3
P25Y0B_An_x03: -0.314160

Weighting of frequency interval 3
P25Y0B_An_w03: 16.000000

Lower bound of frequency interval 4
P25Y0B_An_x04: -0.157080

Weighting of frequency interval 4
P25Y0B_An_w04: 30.000000

Lower bound of frequency interval 5
P25Y0B_An_x05: 0.000000

Weighting of frequency interval 5
P25Y0B_An_w05: 0.000000

Weighting of frequency interval 2
```

P25Y0B_An_w02: 8.000000

Fungal elongation rate on root order 0
PY250FuEl: 0.00020

Depth for root bole cylinder
P25Y0RoBoDp: 0.300000

Radius for root bole cylinder
P25Y0RoBoRa: 0.10000

Appendix D

Root maps used for parameterisation

These maps of 8 and 16 year old *Pinus radiata* root systems where generously made available by Alex Watson, of the Forest Research Institute, New Zealand [62]. Root systems were exposed by sluicing on a sloping site, and roots were plotted in plan and elevation drawings as they were exposed. Eight year old trees 1–4 are plotted with 0.5 m grid markings, the remaining maps are tagged with coordinates in metres. Note that the root editing module of the simulation software corrects for non-square images.

D.1 Eight year old trees

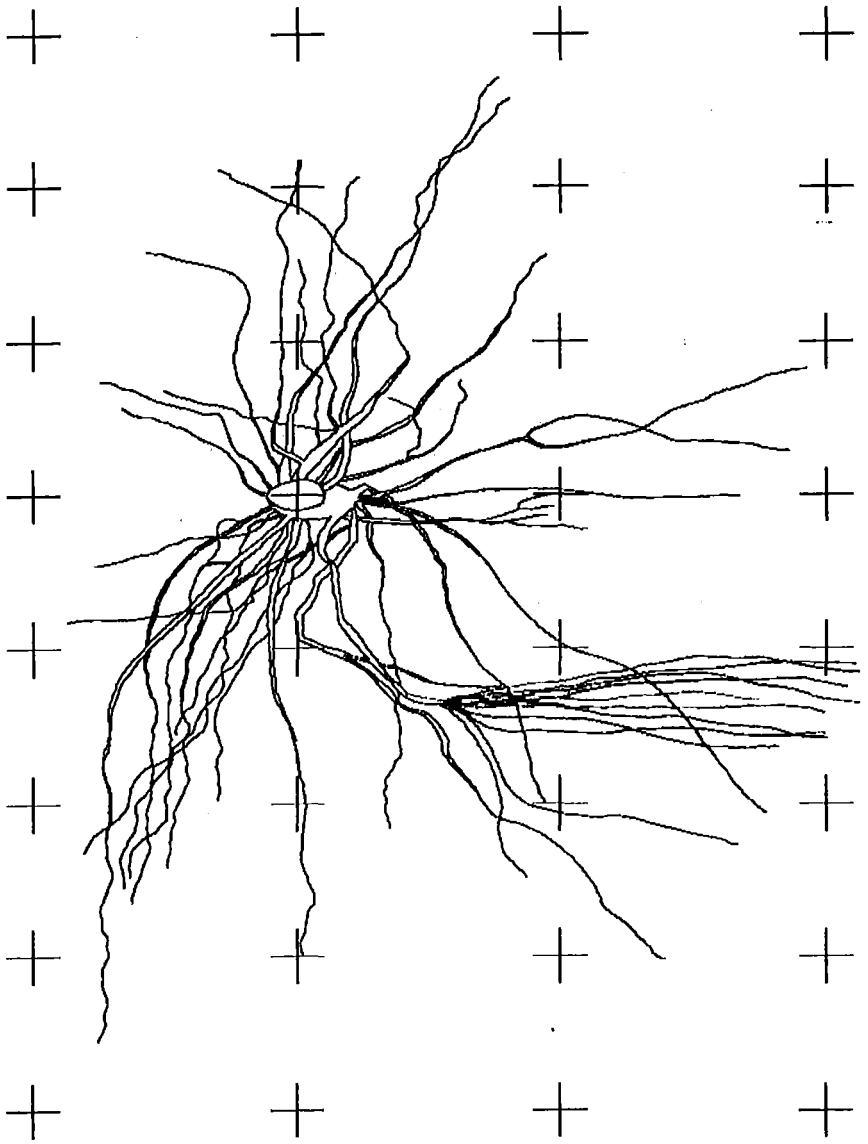


Figure D.1: Tree 8A Plan.

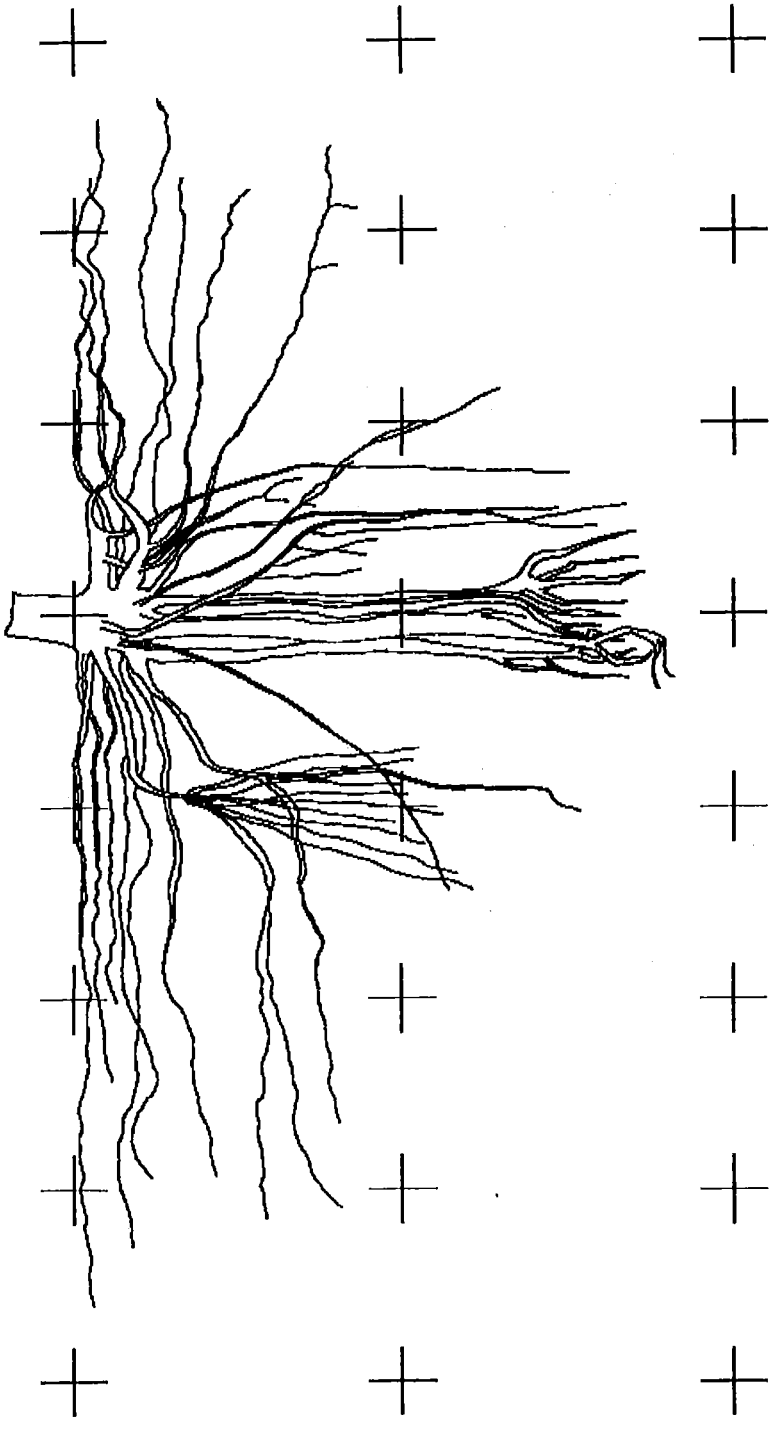


Figure D.2: Tree 8A Elevation.

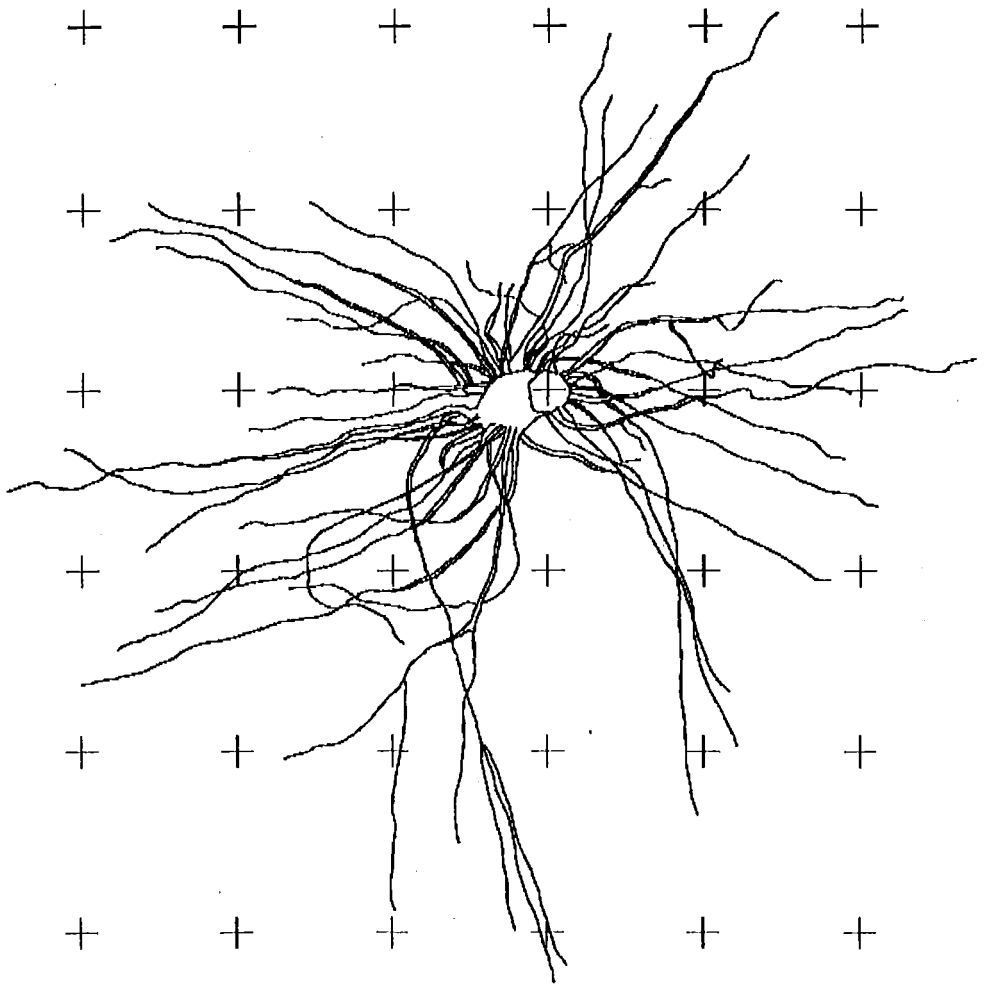


Figure D.3: Tree 8B Plan.

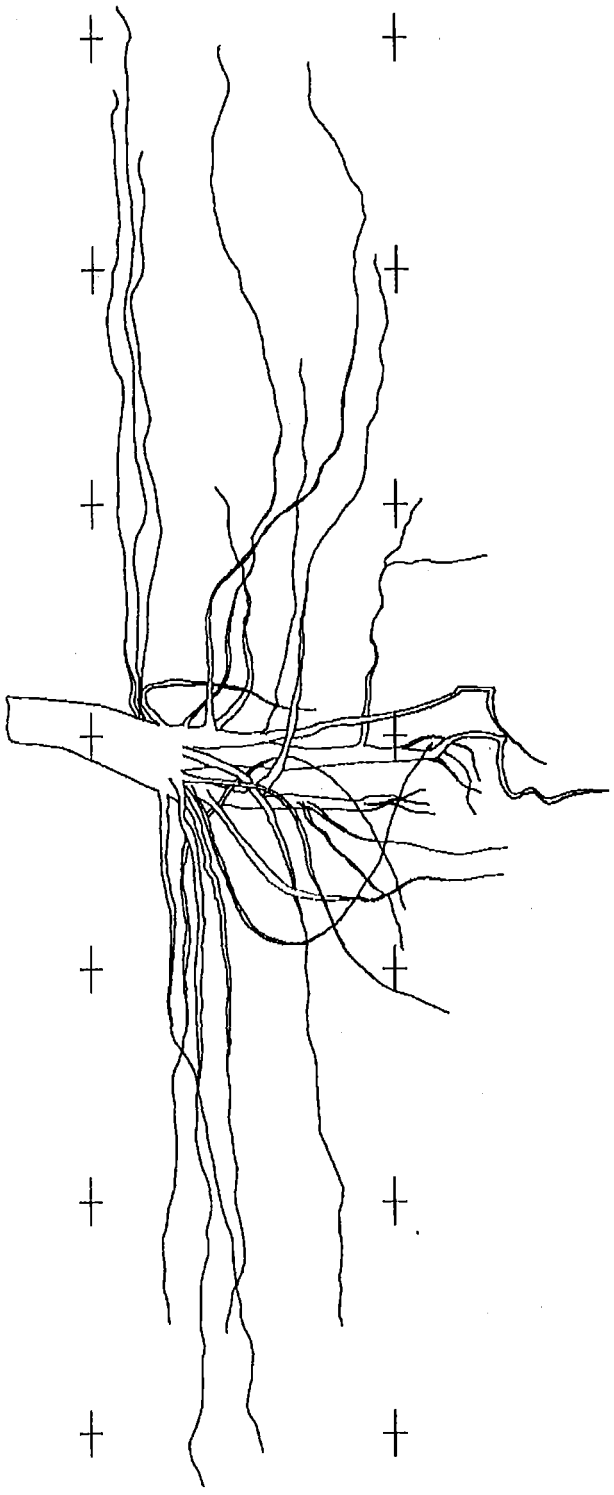


Figure D.4: Tree 8B Elevation.

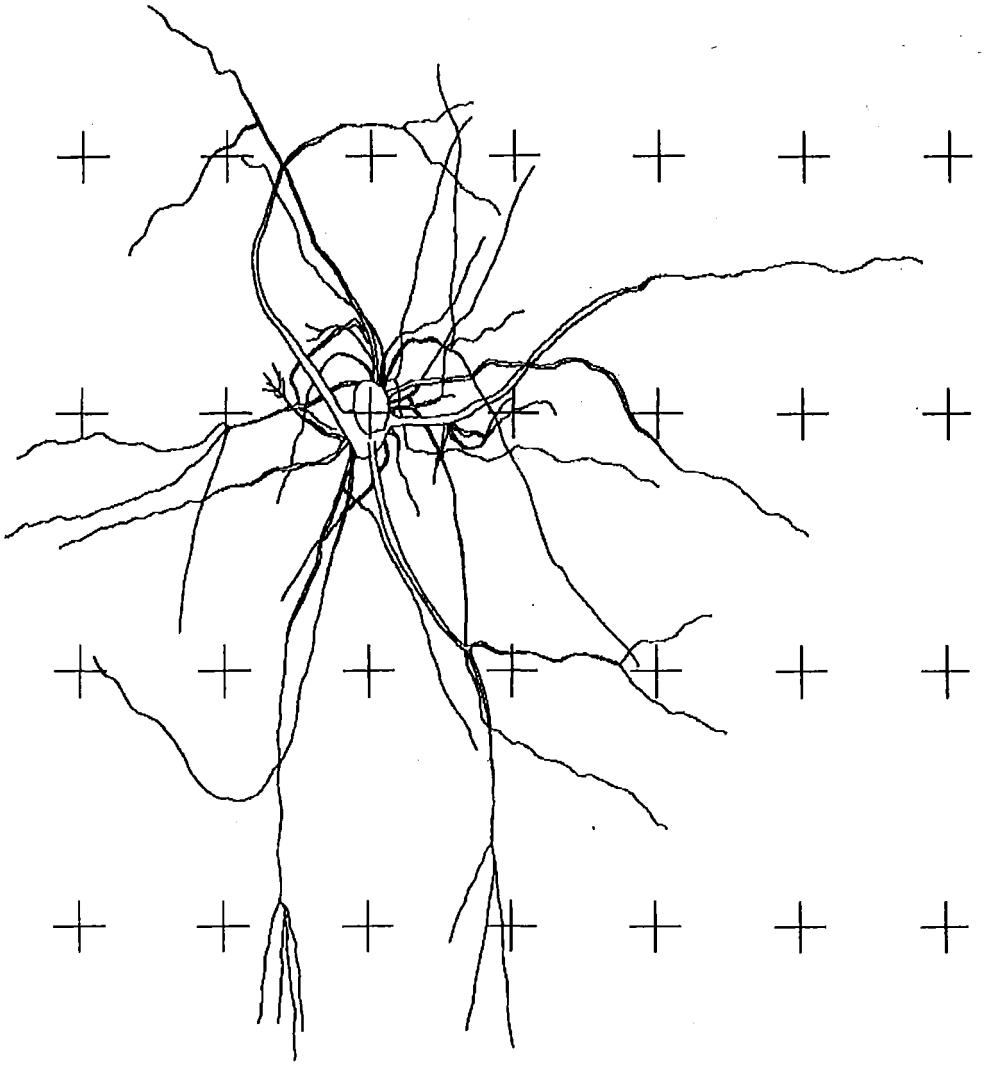


Figure D.5: Tree 8C Plan.

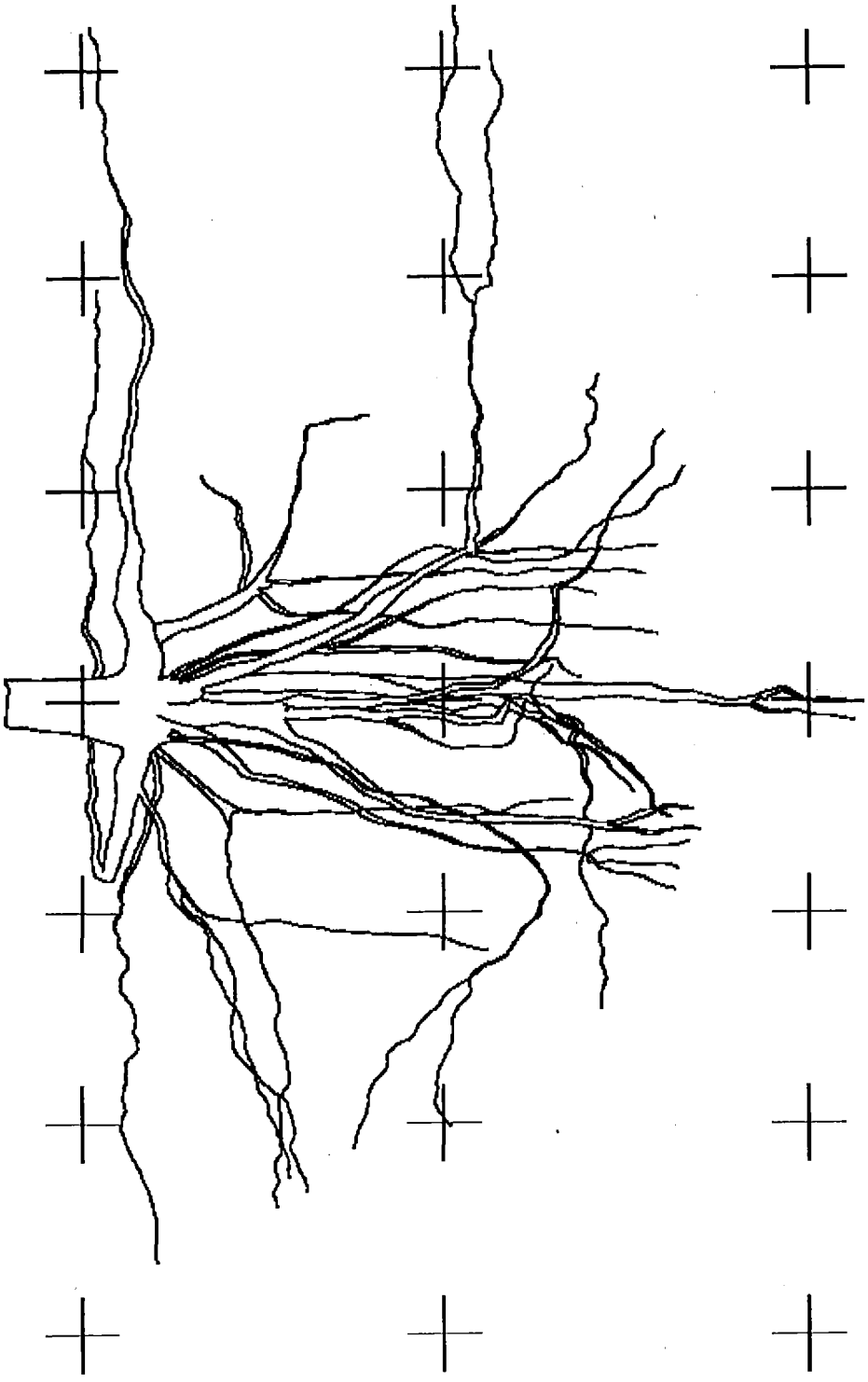


Figure D.6: Tree 8C Elevation.

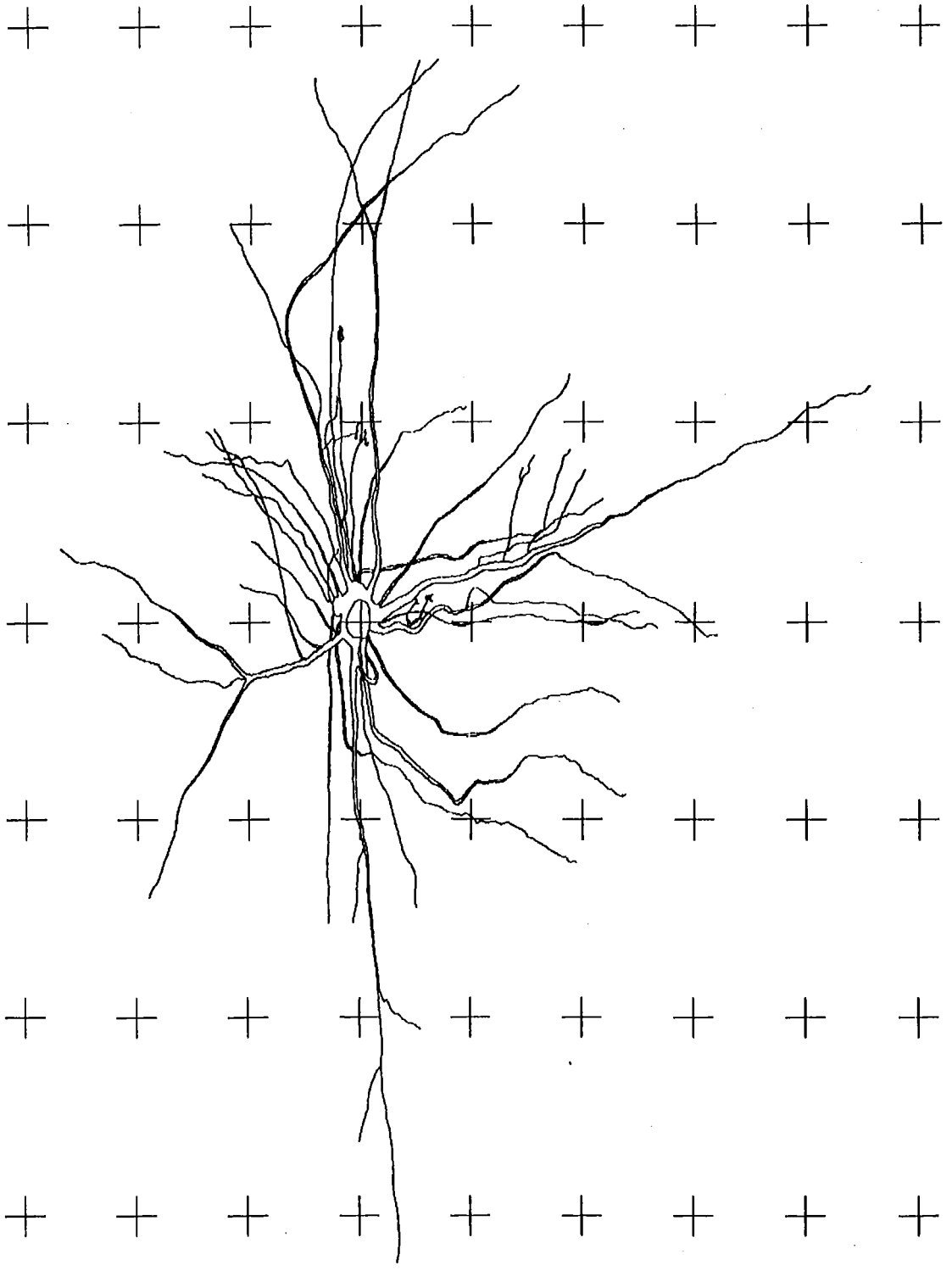


Figure D.7: Tree 8D Plan.

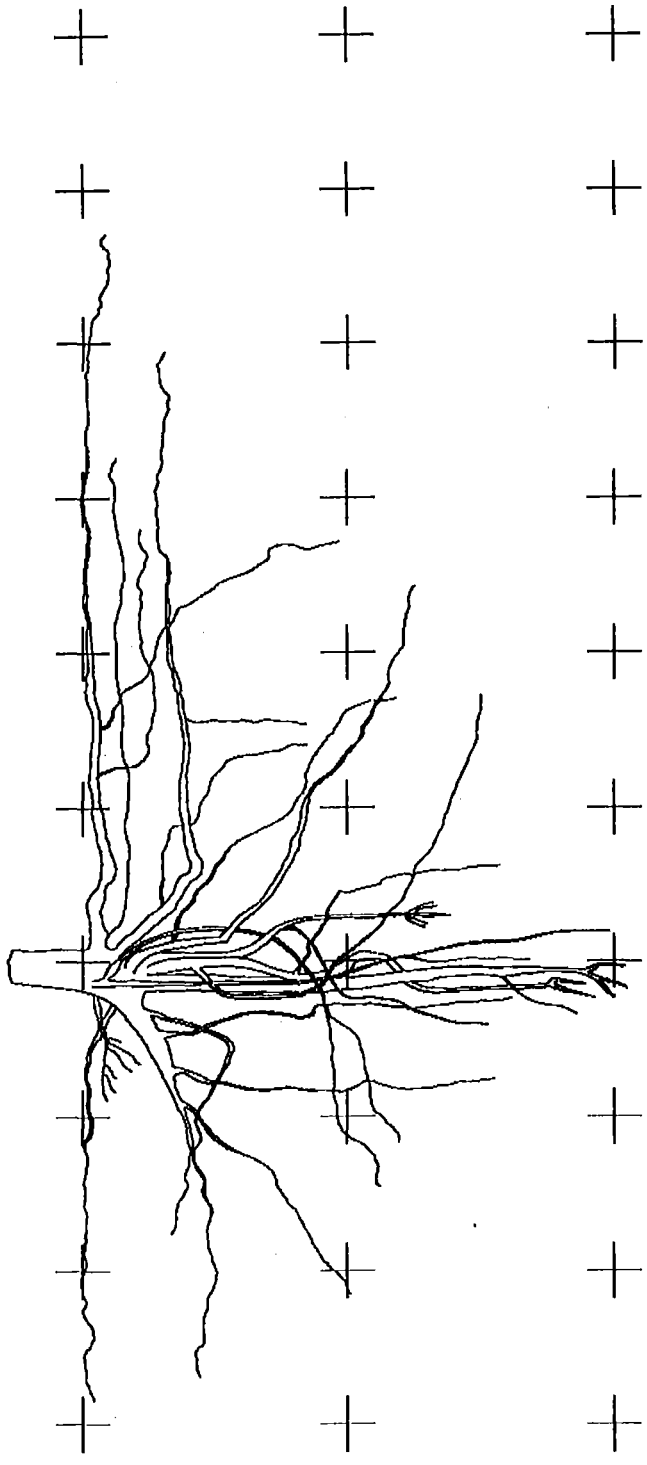


Figure D.8: Tree 8D Elevation.

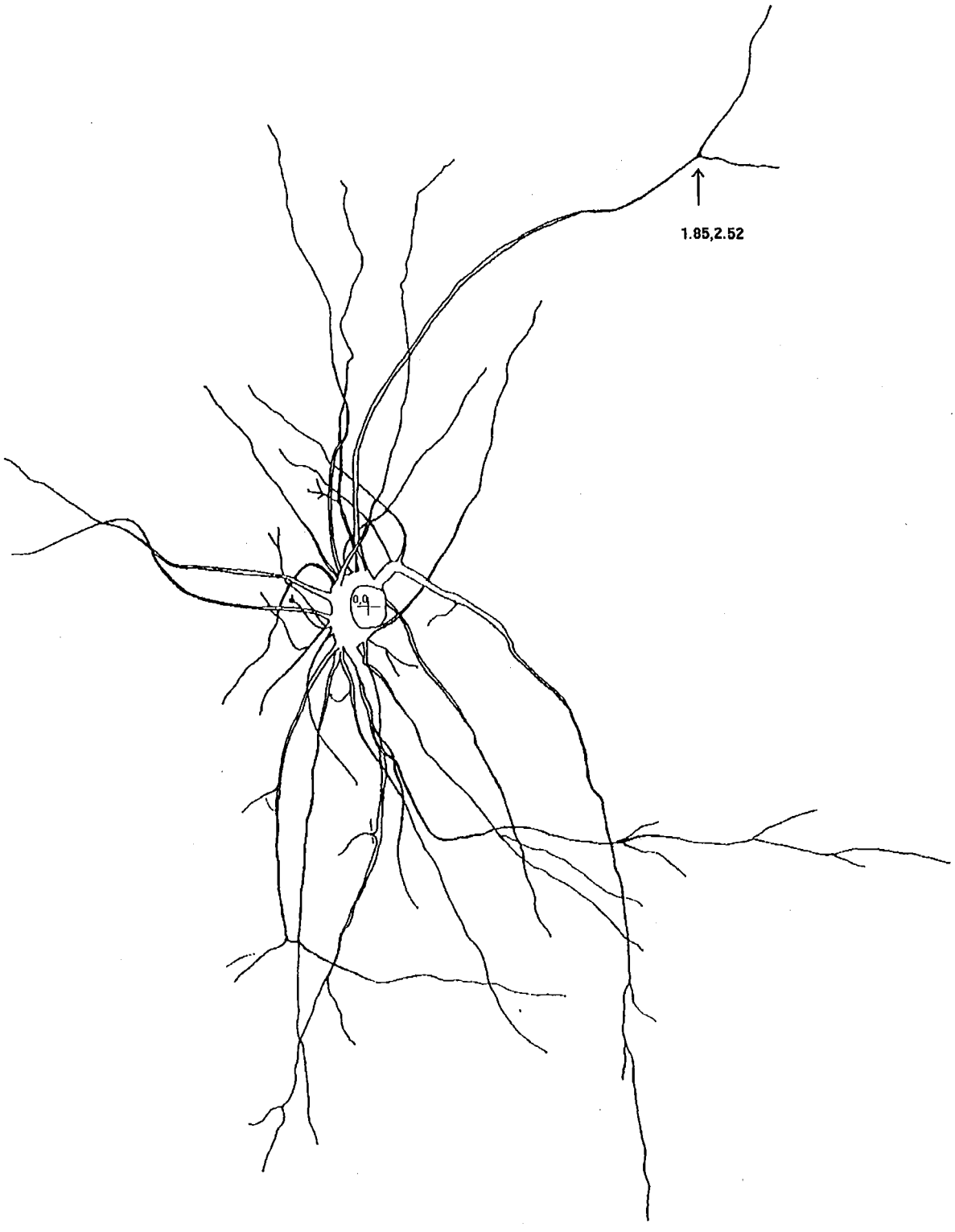


Figure D.9: Tree 8E Plan.

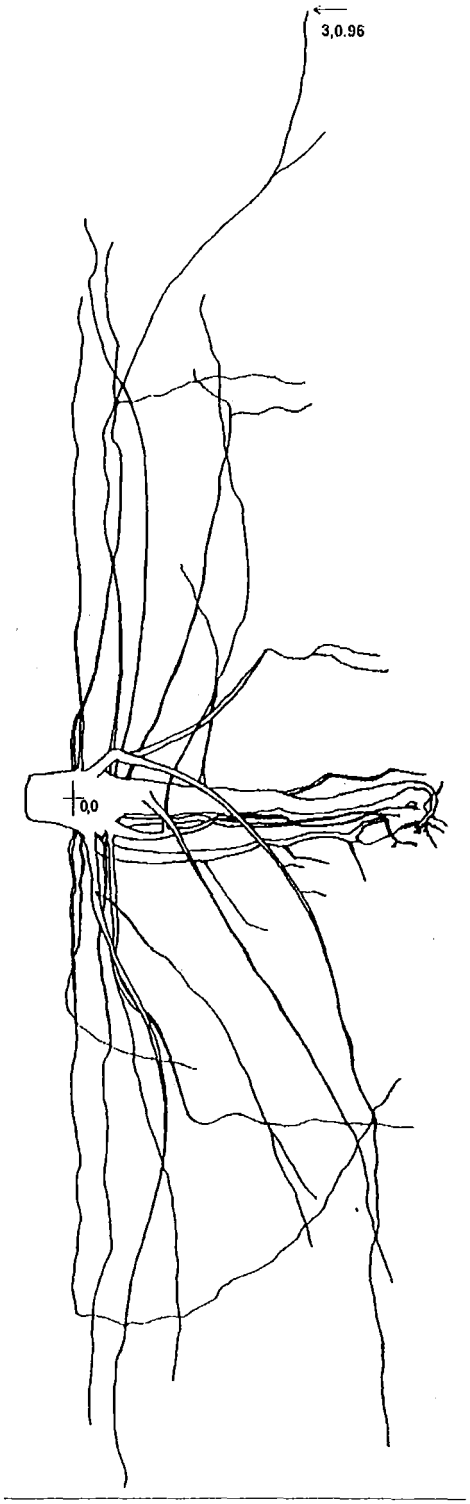


Figure D.10: Tree 8E Elevation.

D.2 Sixteen year old trees

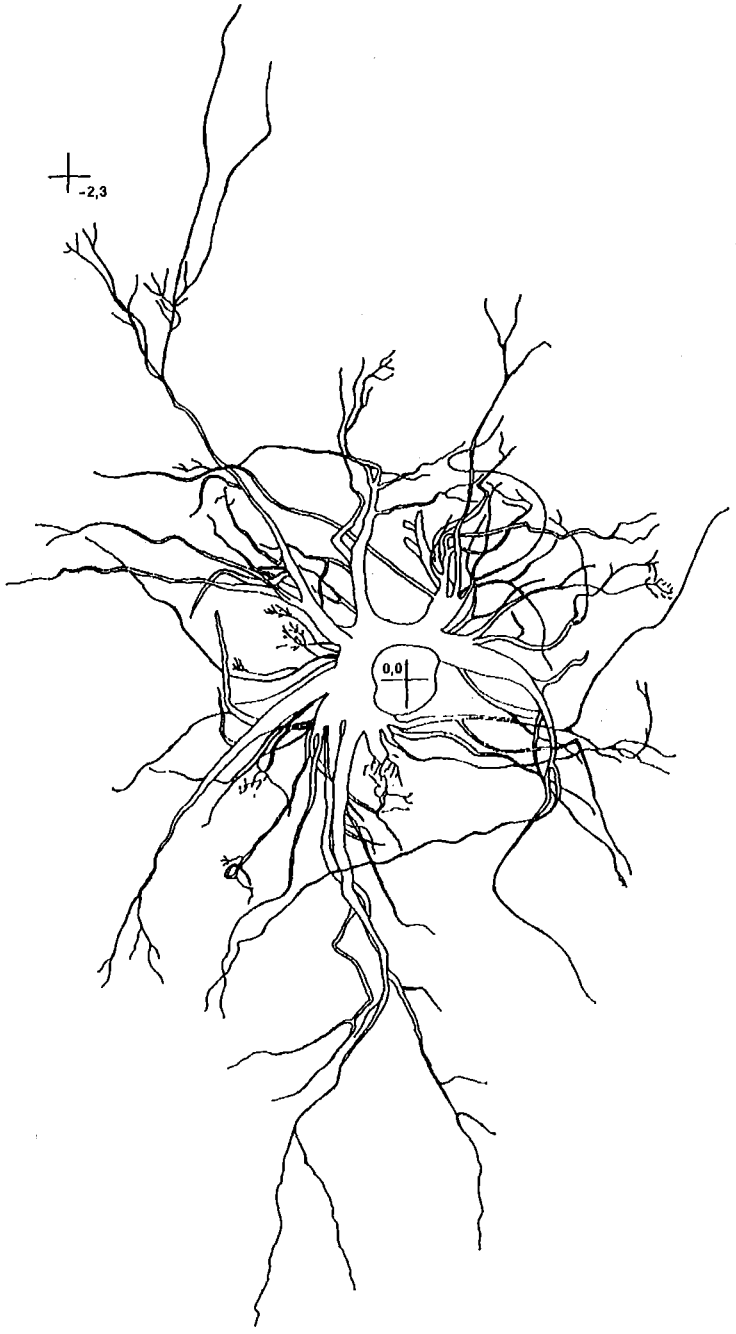


Figure D.11: Tree 16A Plan.

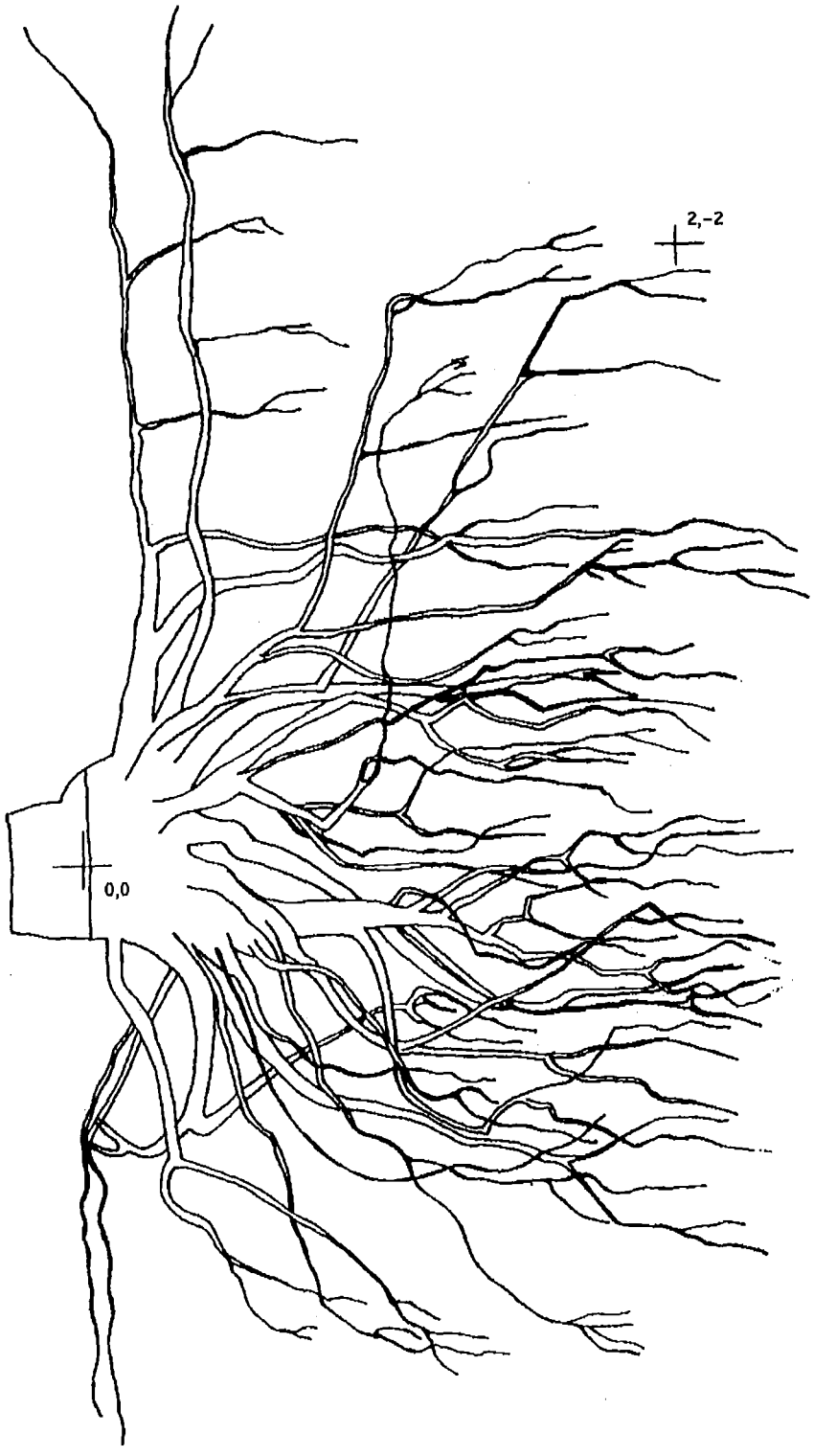


Figure D.12: Tree 16A Elevation.

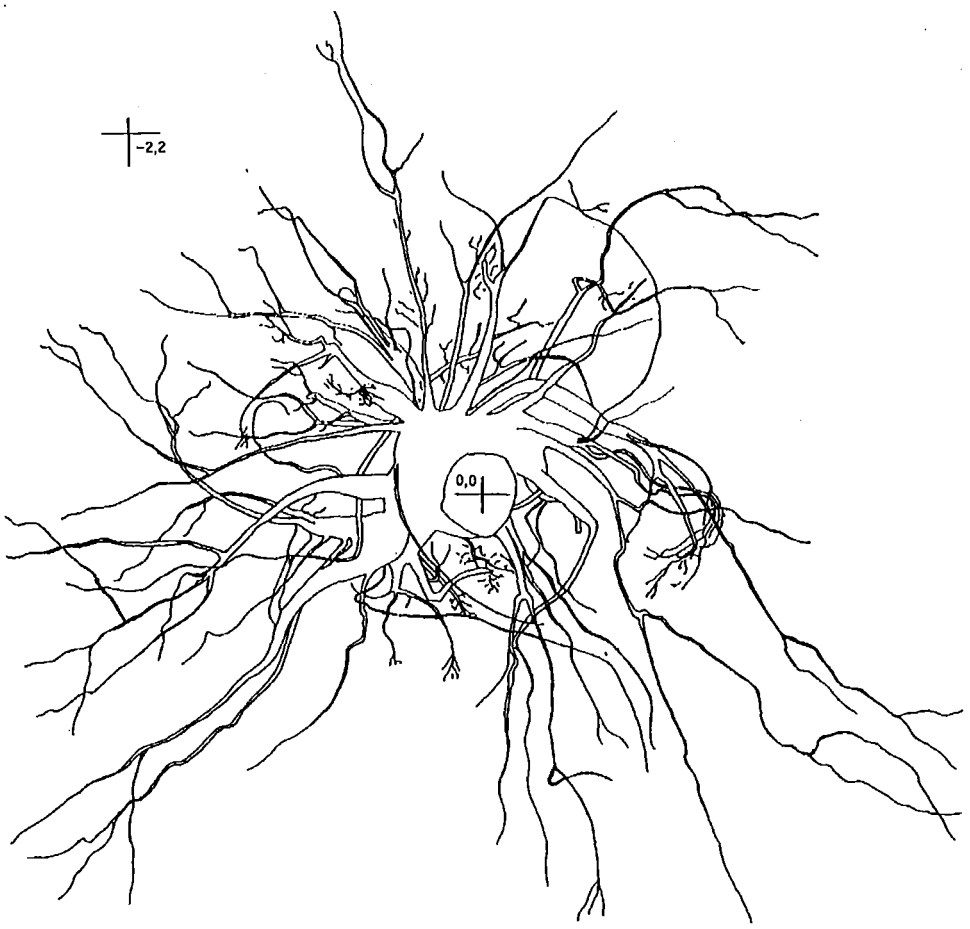


Figure D.13: Tree 16B Plan.

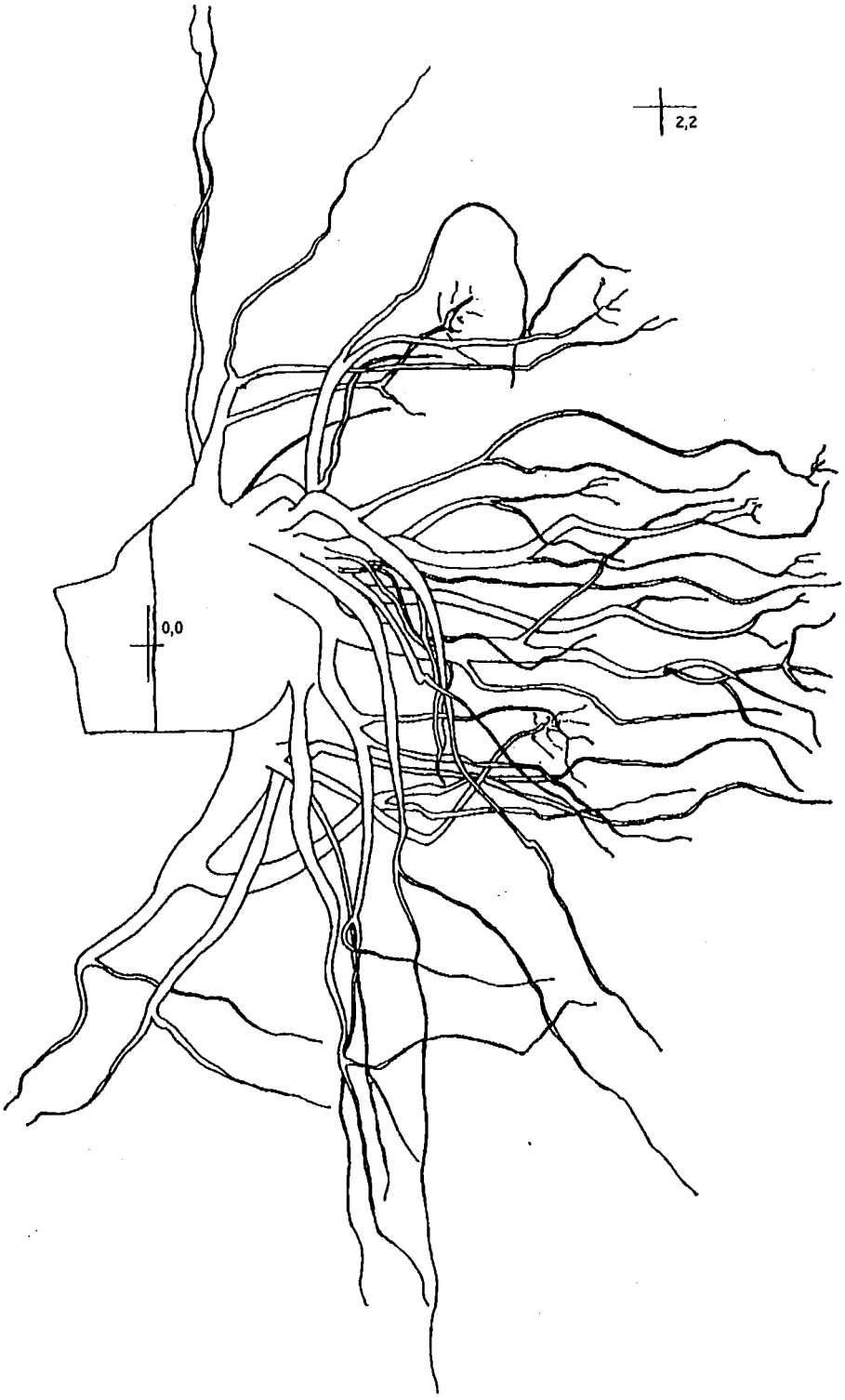


Figure D.14: Tree 16B Elevation.

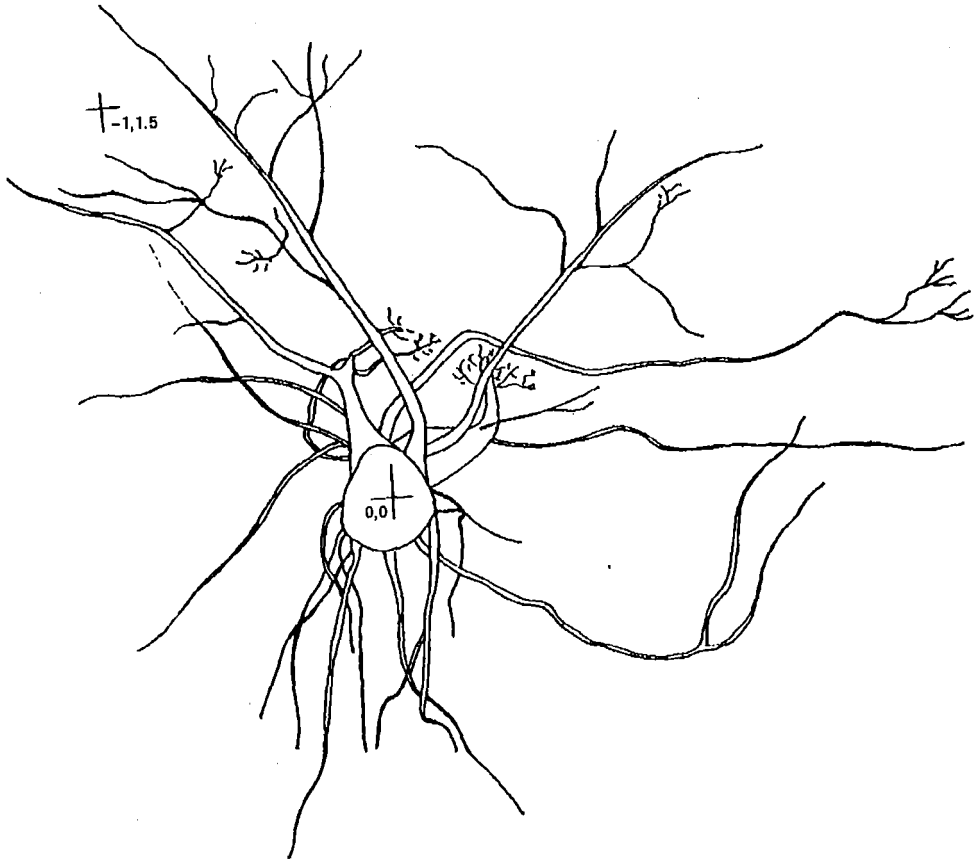


Figure D.15: Tree 16C Plan.

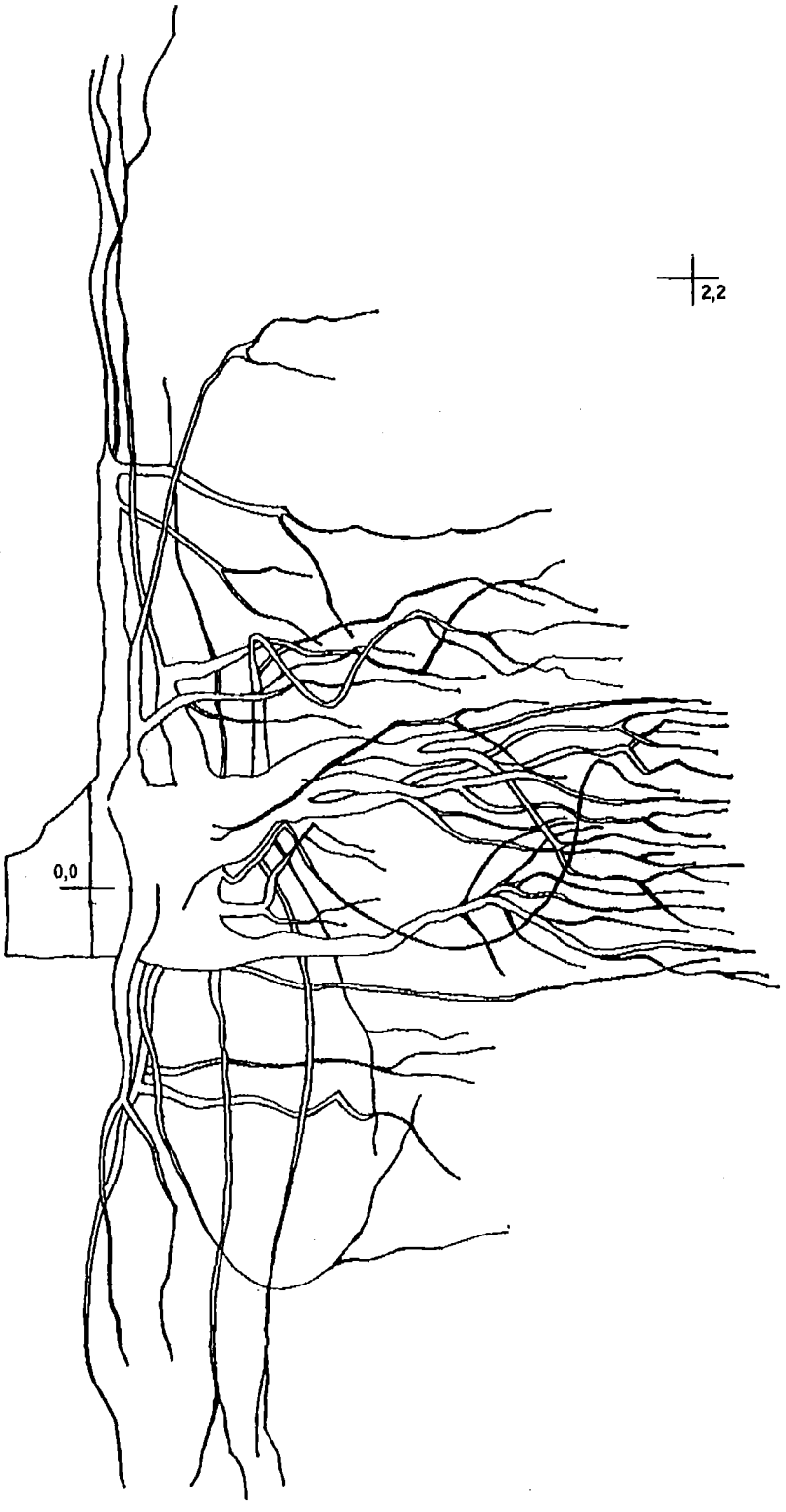


Figure D.16: Tree 16C Elevation.

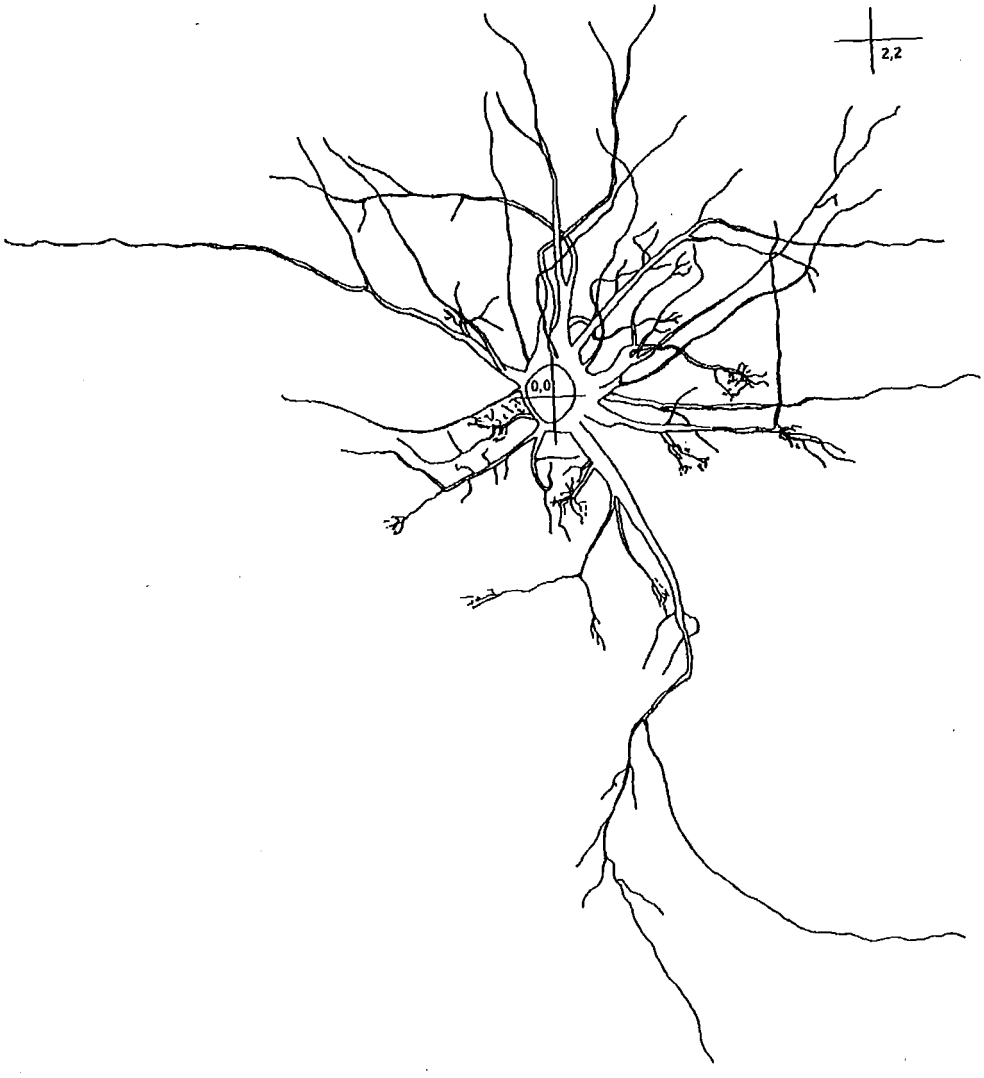


Figure D.17: Tree 16D Plan.

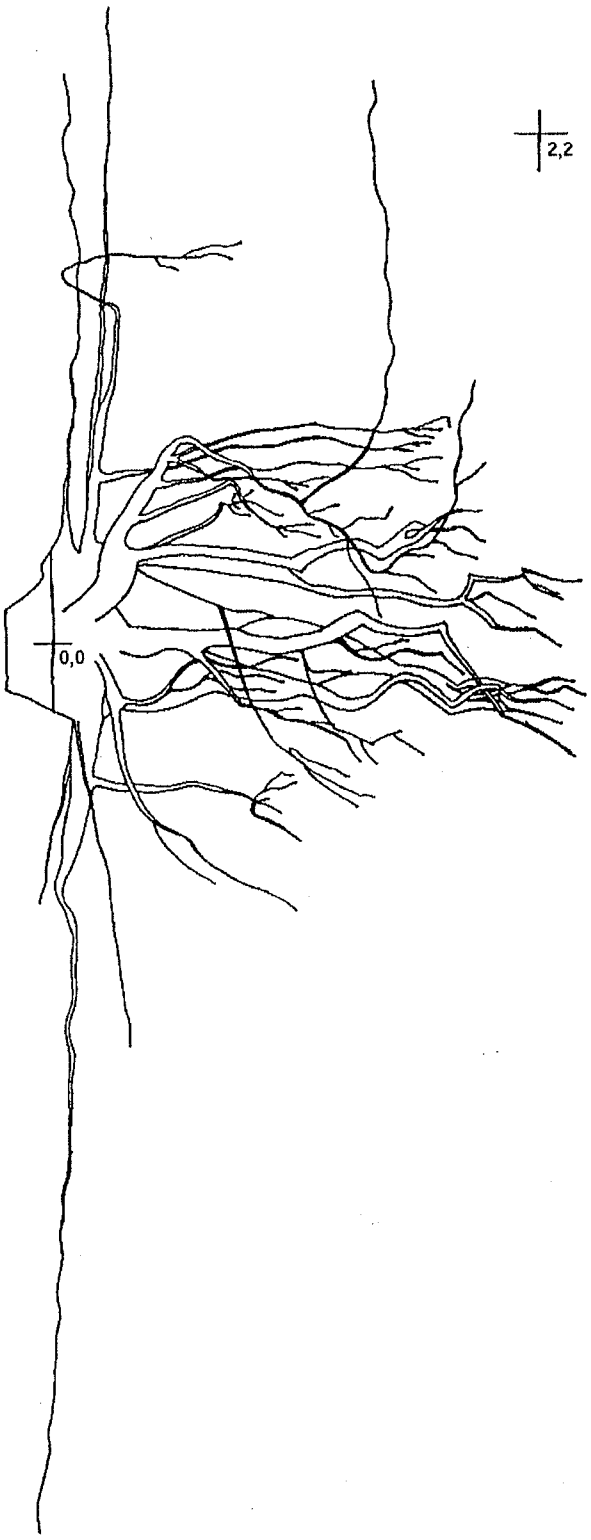


Figure D.18: Tree 16D Elevation.

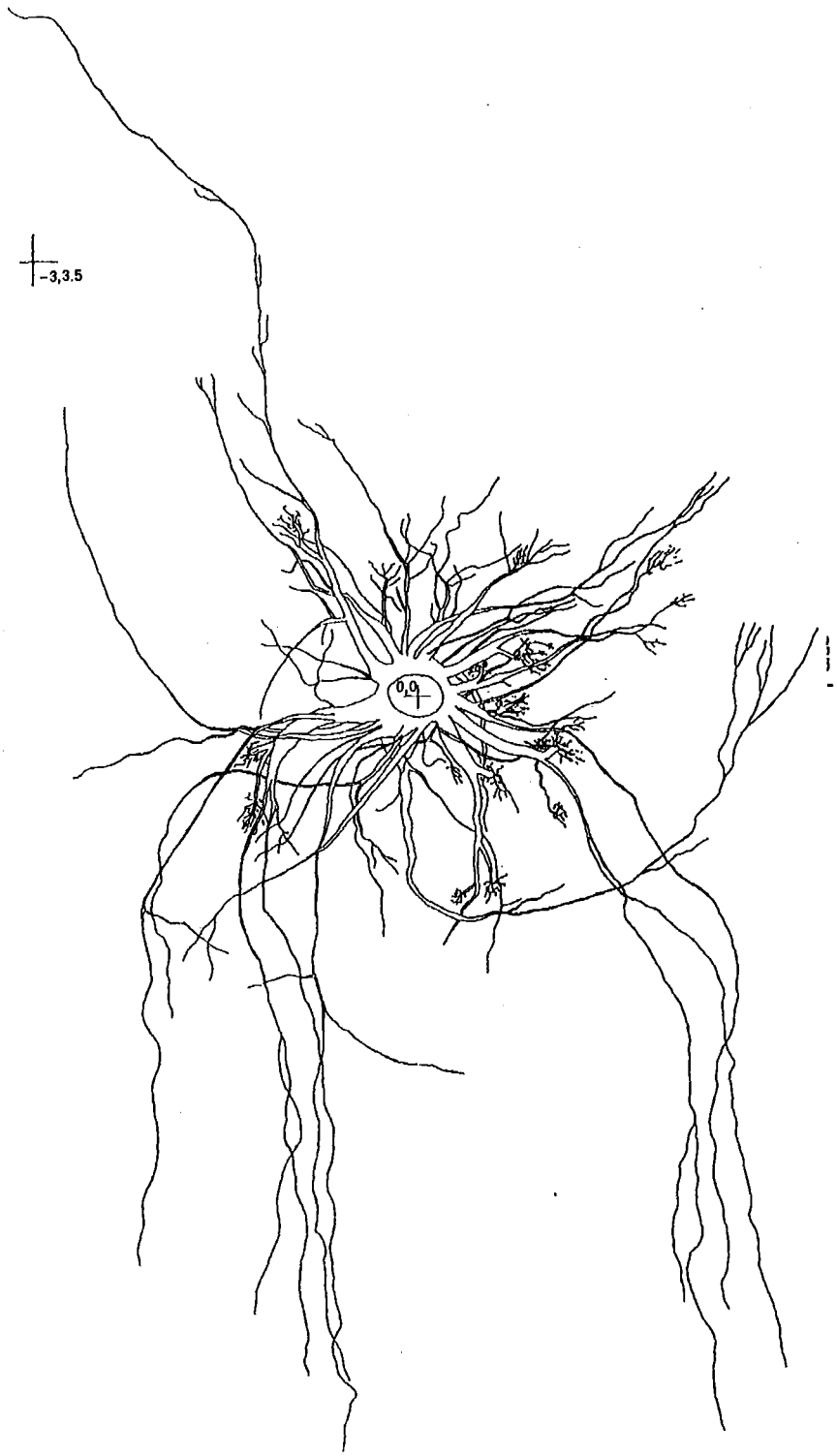


Figure D.19: Tree 16E Plan.

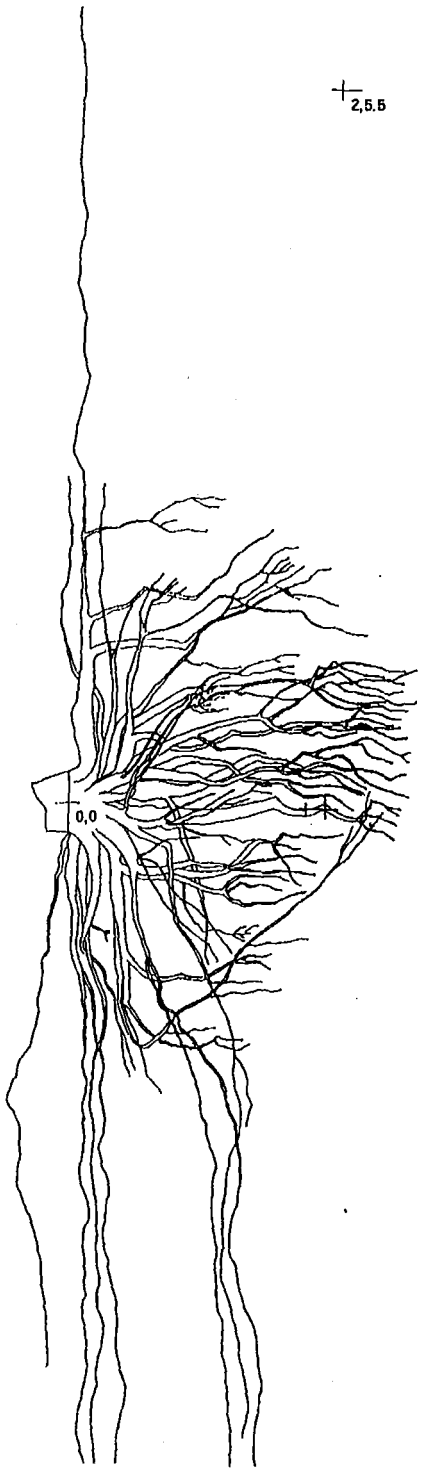


Figure D.20: Tree 16E Elevation.